

Design • Build • Program • Learn • Compete

SERVO

FOR THE ROBOT INNOVATOR
www.servomagazine.com

MAGAZINE
August 2015



**This
Wooden
Hand Can
Teach You**

**Servo Control
Mechatronics
Arduino Programming**

**DARPA
ROBOTICS
CHALLENGE FINALS**

STAY FOCUSED

Capture your world in high definition precision with our pocket-size MD10 action camera and the advanced resolution S60. These exceptional, lightweight cameras keep laser-sharp focus on your multi-servo hexapod maneuvers, exhilarating combot competitions and robotic camp projects. Record and show off your engineering skills by installing one of these visual masters in all your bots!



MD10

FEATURES

- 3 Video Modes: 1080p30, 720p60, WVGA 120fps
- 8 Megapixel Still Images, 8fps Burst
- Intelligent Voice Control
- Optional IP-68 Housing, Waterproof to 20 Meters

TECHNICAL SPECIFICATIONS

- Recording Time (Max.)
Video: About 210 Minutes
Audio: About 240 Minutes
- Dimensions: 1.34 x 1.10 x 2.05 in.
- Weight: 1.76 oz.



S60

FEATURES

- 8 Video Modes: 720p120 ~ 1080p60
- 16 Megapixel Still Images, 8fps Burst
- 2-inch LCD Back for Viewing Footage
- IP-68 Housing, Waterproof to 10 Meters

TECHNICAL SPECIFICATIONS

- Recording Time (Max.)
Video: About 180 Minutes
Audio: About 210 Minutes
- Dimensions: 2.32 x 1.69 x 1.38 in.
- Weight: 1.92 oz.

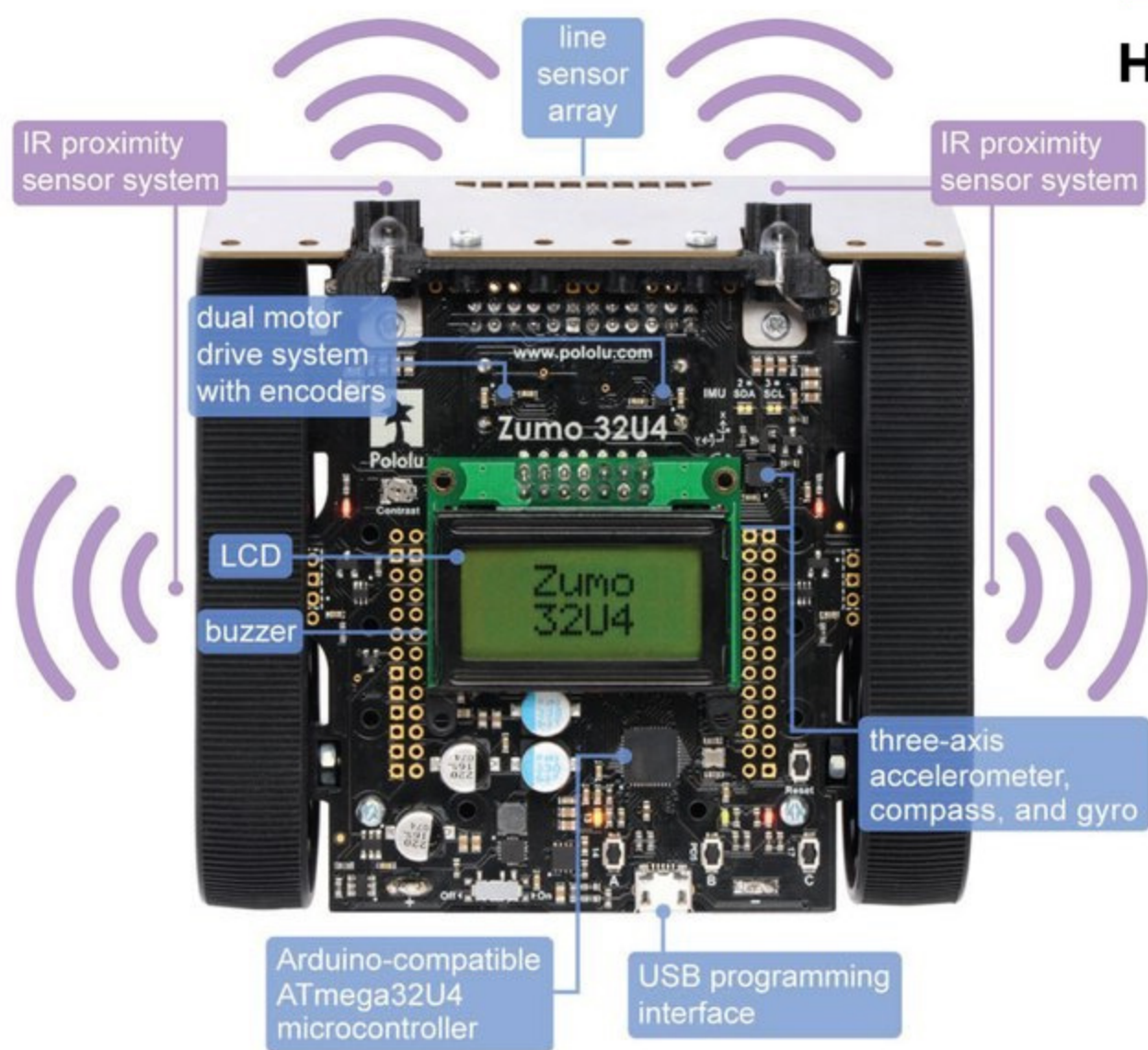


Action Camera
Envision



Zumo 32U4

High performance in an accessible robot



Powerful and versatile, the Zumo 32U4 is a mobile robot accessible to beginners yet engaging for advanced users. The integrated ATmega32U4 offers Arduino-compatible support for those getting started with electronics and microcontrollers, while advanced features include quadrature encoders, a sophisticated object detection system, and an inertial measurement system (IMU), enabling substantial educational and research applications.

Whether you want a competitive Mini-Sumo robot, a robot swarm, or a small friend to roam your desk, the Zumo 32U4 is the robot for you.

Columns

07 Ask Mr. Roboto

by Dennis Clark

Your Problems Solved Here

Adding vision to your automaton.

70 Twin Tweaks

by Bryce and Evan Woolley

The Adventures of Bogie and Sprout

Do some base "jumping" with Runt Rovers — Actobotics newest line of platforms.

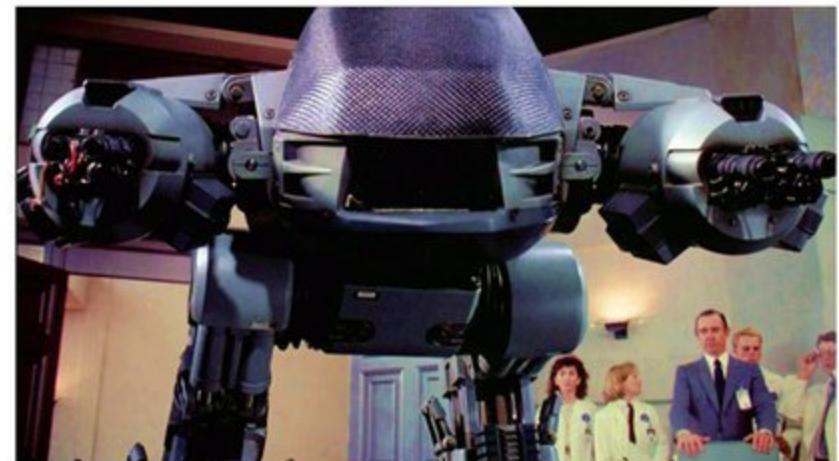
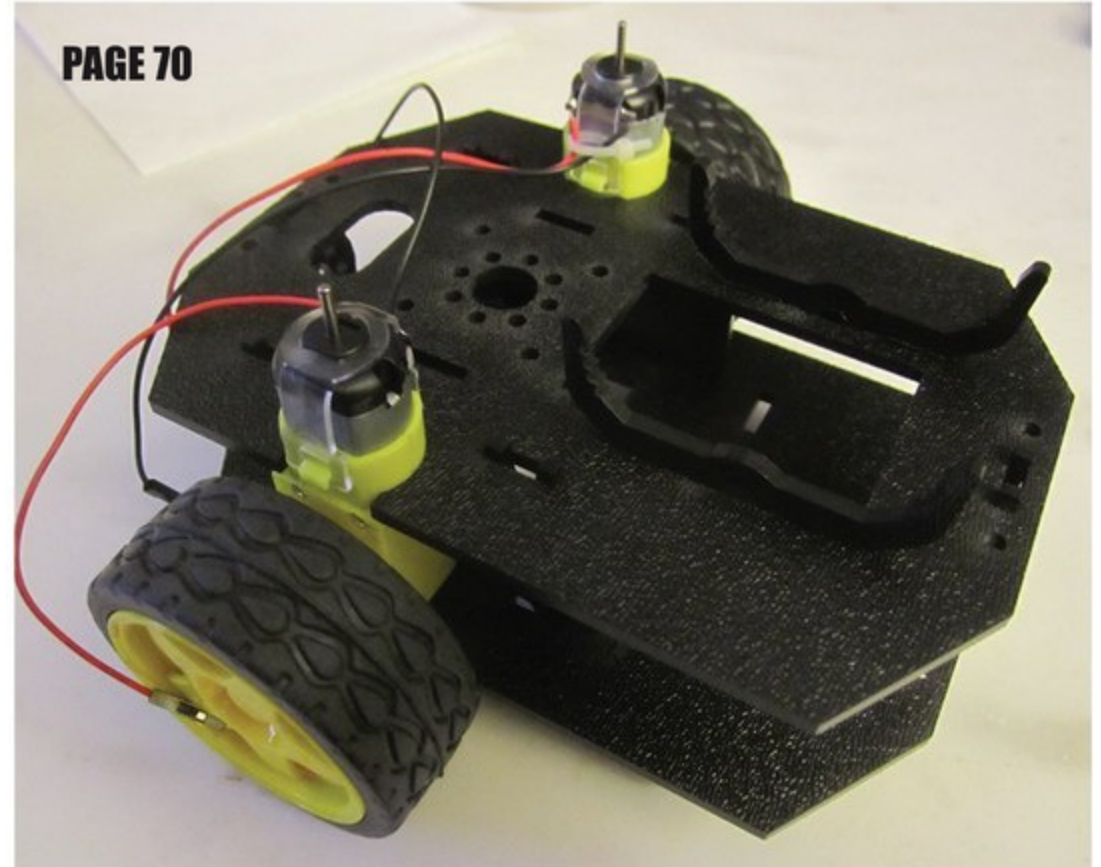
76 Then and Now

by Tom Carroll

Police and Security Robots

With the unrest in today's society and the dangerous kinds of scenarios that are playing out, using robots to protect human life is becoming more and more prevalent. However, you need the right type of robot for the particular job.

PAGE 70



PAGE 76

69 MaxRoboTech Comics

Behaviors, Maps, Landmarks

Departments

06 Mind/Iron

Not Your Grandmother's Singer

10 New Products

11 Showcase

15 Events Calendar

66 SERVO Webstore

81 Appetizer

Battling Robots. Are they violent?

82 Robo-Links

82 Advertiser's Index

12 Bots in Brief

- Drone to the Rescue
- Folds Under Pressure
- Google's Driving Record
- We Be Jammin'
- Towing AVERTed
- Mid-Air Hookups





27 Little Arduino Robot Hand

by Peter Ohlmus

Although this wooden hand looks simple, it demonstrates some fundamentals of servo control, Arduino programming, and mechatronics for under \$300.

36 Animatronics for the Do-It-Yourselfer

by Steve Koci

It's All in Your Head!

Make your robotic characters more life-like by giving them a voice and some head movement with these simple-to-learn techniques.

45 Beer2D2 — A Robot in a Can

by Steven Nelson

A chronicle of how an old lager was reborn as a loyal friend and show companion.

52 CNC Part Creation Workflow

by Michael Simpson

This time, see how to use G-code files to create actual parts.

58 The DARPA Robotics Challenge Finals 2015

by Thomas Messerschmidt

Get a bird's eye view of this "Super Bowl" of robotics.

The Combat Zone

16 Back to Do Battle

18 Building Better Bots: The Weak Link

20 BUILD REPORT

The Build of Ash and Its First Event

23 EVENT REPORT

Freeside Robot Street Fight 2015



Not Your Grandmother's Singer

The current poster child of the DIY community is the 3D printer. However, from both a technological and market share perspective, the modern inexpensive sewing machine is the creative DIY tool of the masses. For the price of three or four spools of ABS plastic filament, you can purchase a computerized, self-threading sewing machine that can semi-autonomously create everything from button holes and decorative stitching to piping and rolled edges. Not only can these machines create a variety of stitches, but they can automatically adjust to materials of different weight and thickness. All of this automation translates to greater ease of use, a shorter learning curve, and more creative possibilities.

As with a typical 3D printer, modern sewing machines contain multiple motors — each under microcontroller control — and a variety of sensors. This means you can opt for full power at super slow speeds — great for learning or working with thick materials. Instead of changing print heads to handle different types of extrusion materials, you can change the small metal feet to work with different cloth, leather, or soft metals. For example, there are feet for making piping, for sewing on zippers, and for sewing especially fragile fabrics.

I learned to sew when computerized machines weren't even on the drawing boards, which probably explains why my go-to machine for straight stitching is a 75 year old Singer 201-2. However, I've supplemented this 40 lb, US made chunk of iron with a modern, computerized (and mostly plastic) 1034D Serger by Brother. Together, I can handle just about any DIY challenge that lends itself to sewing.

My current sewing projects include developing wearable sensor networks, using conductive threads, waterproof sensors, and displays. I use the piping attachment on my Serger to create an insulated conduit for battery power and decorative metallic thread for short, direct connections. For microcontrollers and sensors, I use the Arduino family of wearable components from AdaFruit and SparkFun.

I've had good success with FLORA — the Arduino-comparable wearable electronic platform (Adafruit, \$20). A few years ago when I started experimenting with wearable sensor networks, I used silver-impregnated thread to connect sensors to a predecessor of FLORA. This was problematic, in that I couldn't wash the garment without increasing the resistance of the thread due to lost silver. Furthermore — even when new — the silver thread had considerable resistance. Anything over about a foot was simply an exercise in frustration. Today, I use stainless steel thread to connect the FLORA microprocessor to peripherals and battery power located anywhere on the body. Because it's not insulated, I usually tuck the wire in vinyl piping or hollow tube that looks like normal trim to a casual viewer.

So, if you haven't checked out a sewing machine in a few years, you owe it to yourself to play with one of the newest computerized models. Take a look at the Brother CS6000i — the best seller on Amazon. It offers an impressive array of features for less than \$150. Who knows, you just might reenergize the DIYer inside you. **SV**

FOR THE
ROBOT
INNOVATOR

SERVO
MAGAZINE

Published Monthly By

T & L Publications, Inc.

430 Princeland Ct., Corona, CA 92879-1300

(951) 371-8497

FAX **(951) 371-3052**

Webstore Only **1-800-783-4624**

www.servomagazine.com

Subscriptions

Toll Free **1-877-525-2539**

Outside US **1-818-487-4545**

P.O. Box 15277, N. Hollywood, CA 91615

PUBLISHER

Larry Lemieux

publisher@servomagazine.com

**ASSOCIATE PUBLISHER/
ADVERTISING SALES**

Robin Lemieux

robin@servomagazine.com

EDITOR

Bryan Bergeron

techedit-servo@yahoo.com

VP of OPERATIONS

Vern Graner

vern@servomagazine.com

CONTRIBUTING EDITORS

Tom Carroll

Kevin Berry

Dennis Clark

R. Steven Rainwater

Steven Nelson

Peter Ohlmus

Mike Jeffries

Steve Koci

Michael Simpson

Andrea Suarez

Bryce Woolley

Evan Woolley

Russ Barrow

Jack Tweedy

Thomas Messerschmidt

CIRCULATION DEPARTMENT

subscribe@servomagazine.com

WEB CONTENT

Michael Kaudze

website@servomagazine.com

WEBSTORE MARKETING

Brian Kirkpatrick

sales@servomagazine.com

WEBSTORE MANAGER

Sean Lemieux

ADMINISTRATIVE STAFF

Debbie Stauffacher

Re Gandara

Copyright 2015 by

T & L Publications, Inc.

All Rights Reserved

All advertising is subject to publisher's approval. We are not responsible for mistakes, misprints, or typographical errors. *SERVO Magazine* assumes no responsibility for the availability or condition of advertised items or for the honesty of the advertiser. The publisher makes no claims for the legality of any item advertised in *SERVO*. This is the sole responsibility of the advertiser. Advertisers and their agencies agree to indemnify and protect the publisher from any and all claims, action, or expense arising from advertising placed in *SERVO*. Please send all editorial correspondence, UPS, overnight mail, and artwork to: **430 Princeland Court, Corona, CA 92879.**

Printed in the USA on SFI & FSC stock.



ASK MR. ROBOTO

Your robotic problems solved here.

by Dennis Clark

Our resident expert on all things robotic is merely an email away.

roboto@servomagazine.com

Tap into the sum of *all human knowledge* and get your questions answered here! From software algorithms to material selection, Mr. Roboto strives to meet you where you are — and what more would you expect from a complex service droid?

Post comments on this article at www.servomagazine.com/index.php/magazine/article/august2015_MrRoboto.

Mr. Roboto is limping back from crawlspace floodwaters this month. The "mile high" land of Colorado seems more like a mile deep these days. My spirits are high, however, and I have nearly beaten the kelpies back from taking over the house. Things should be settling down soon! As I write this column, the SparkFun 7th annual AVC competition is coming up in a week. You can bet Mr. Roboto will be there to cheer everyone on and watch for innovations that we can all pick up on! Anyway, let's move onward to this month's topic: robot vision.

I have been talking about adding vision to robots these last two columns and I will continue with that theme this month. There are many camera options to choose from that are small enough to fit on just about any robot. The simple camera modules often output their data in ways that our lower cost controllers cannot handle. Some output analog levels need to be converted on a pixel-by-pixel basis, and we need to track horizontal and vertical sync signals. Some output digital values for each pixel, but use a compressed video format that our microcontrollers lack the processing power (or memory space) to deal with.

With that in mind, I went searching for cameras that could be interfaced to our simple microcontrollers that can be used with rudimentary vision algorithms to do either navigation or object detection.

The Gameboy Camera Hack

When we consider using inexpensive vision systems with our inexpensive robot controllers, we should mention the first hacks which made this possibility a reality over 10 years ago. I speak, of course, about the Gameboy® camera hacks that were able to give small robots monochrome vision. The first hack that I know about is detailed in the *SRS Encoder* (Seattle Robotics Society Newsletter) in 2002 (www.seattlerobotics.org/encoder/200205/gbcam.html).



Figure 1.

The article was by Dafydd Walters and used a 68332 robot board. The link will give you the details of how he used the camera.

The second source I know personally. Daniel Harrington showed me his Gameboy camera hack in action. Daniel used a custom controller board with an Atmel 8515 that he demonstrated following a white ping-pong ball. A write-up of his design is actually on the Atmel microcontroller site.

I tried to include the direct link here, but the URL is *really* ugly. So, I suggest that you go to www.atmel.com and search for "issue4_pg39_43_robotics.pdf" to get Daniel's project. **Figure 1** is a picture of his ping-pong ball tracking robot. His article can also be found in *Circuit Cellar* issue #151 (February 2003); you can find his code files at www.dtweed.com/circuitcellar

[/caj00151.htm](http://www.dtweed.com/circuitcellar/caj00151.htm).

This camera is particularly interesting because it had edge-detection features that simplified using it for object tracking. Because it is monochrome, you can pull an image from it with our newer, faster microcontroller boards very quickly to analyze images in real time.

If you like hacking your own stuff, I recommend that you check these articles out and try it yourself. You can still get these cameras on eBay for \$7 or so. The total cost of this project will be time and probably less than \$10 in parts (not including the microcontroller board).

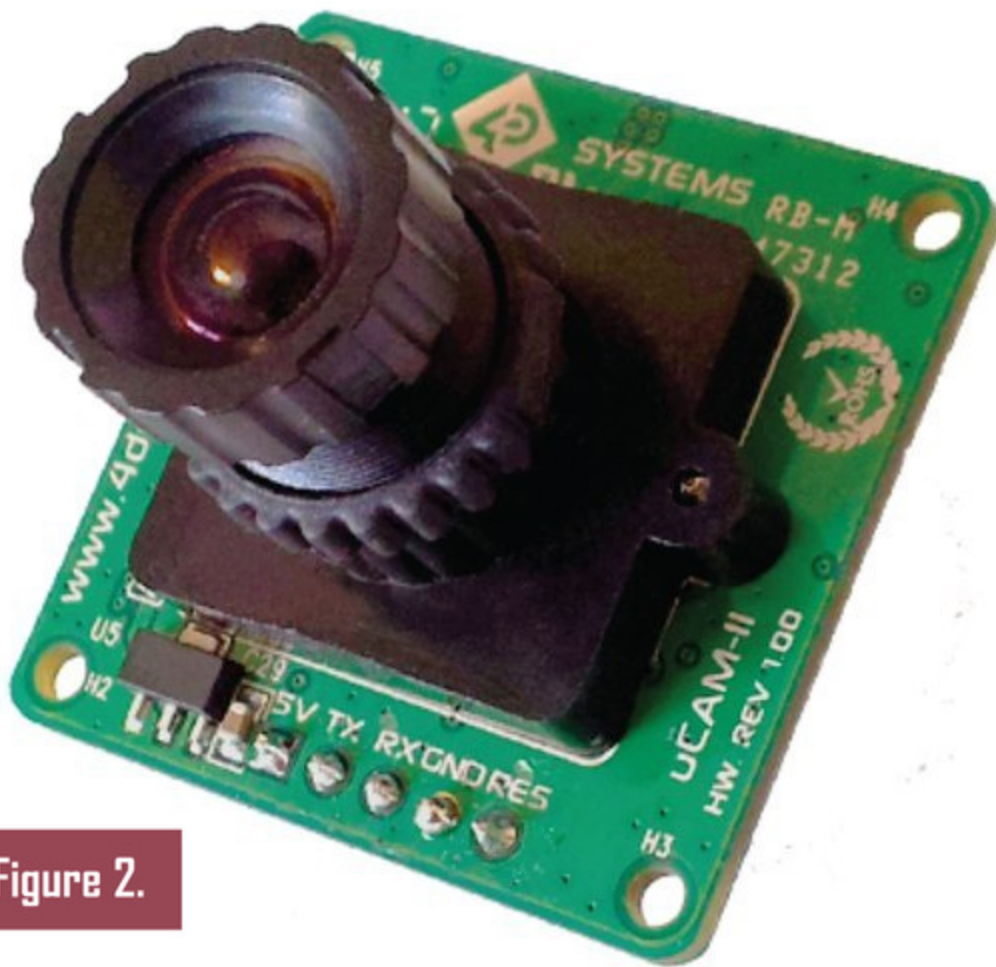


Figure 2.

has its Workshop 4 IDE (free) to help you to configure your camera board very easily. Go to www.4dsystems.com.au/product/uCAM_II/ for more details.

The CMUCam5 Pixi

The next vision board to consider is the venerable CMUcam series. CMUcam's 1-4 have been retired and replaced by the newest in the line: the Pixi CMUcam5. Smaller, better, faster (one assumes; **Figure 3**). CMUcams have object tracking software built into them, and also nifty Java based tools to use on your computer to help you configure and test the camera.

This camera can output results at 50 fps, so tracking speed is definitely adequate for robot navigation experiments. Not only that, this vision board doesn't just output images, it tells you where things are and even has a built-in pan/tilt function for servos. The CMUcam5 can be purchased in many places; check out their website at www.cmucam.org/. As you can also see, this is an open source project with lots of users to help you on your way. The cost of a CMUcam5 is from \$70 to \$85, depending on the merchant.

Arducam Shield and Arducam Mini

Because you can't seem to talk about robots without talking about Arduino, there is the Arducam series of Arduino compatible cameras. There are two that I know of: the Arducam Shield Rev. B or Rev. C (www.arducam.com/tag/arducam-shield-2/) which come with a bewildering set of options. One nice feature of this board is a built-in uSD card slot for image storage. The other nice feature is the camera data SPI interface has a maximum clock speed of 8 MHz. This means that a 320x240 RGB(565) screen of 76K bytes can stream at about 13 fps. The Arducam Shield is an open source project, so full schematics and code is available to users and hackers to play with. It was not easy to find documentation for this shield, but Google came to the rescue: www.arducam.com/category/user-guide/. The documentation is not extensive, so there will be some challenge to your project!

The Arducam Shield is simply an interface to a camera; there is no processing done on the shield. That project is left to the program running on the microcontroller board. See **Figure 4** for an image of the shield sans camera module. The Arducam site points to the UCTronics site for purchasing them (www.uctronics.com/catalogsearch/result/?q=arducam+shield), where the shield without a camera costs \$29.99. Camera modules appear to run from \$6 to \$13 on that site. Or, you could go to Marlin P. Jones and get their entry for \$39.95, which comes with a two megapixel camera installed (www.mpja.com/Camera-Arduino-Compatible-Shield/productinfo/31065%20MP).

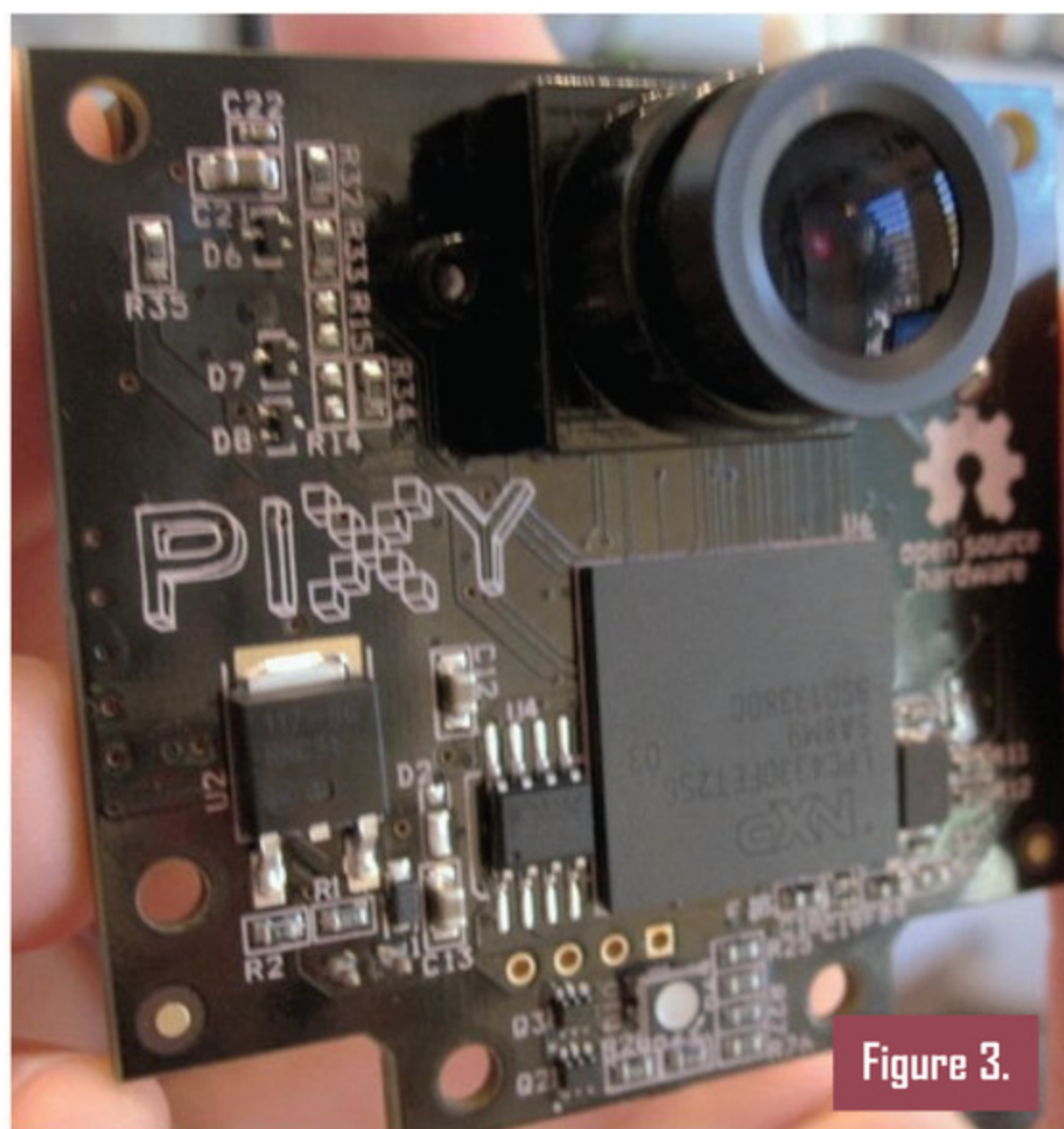
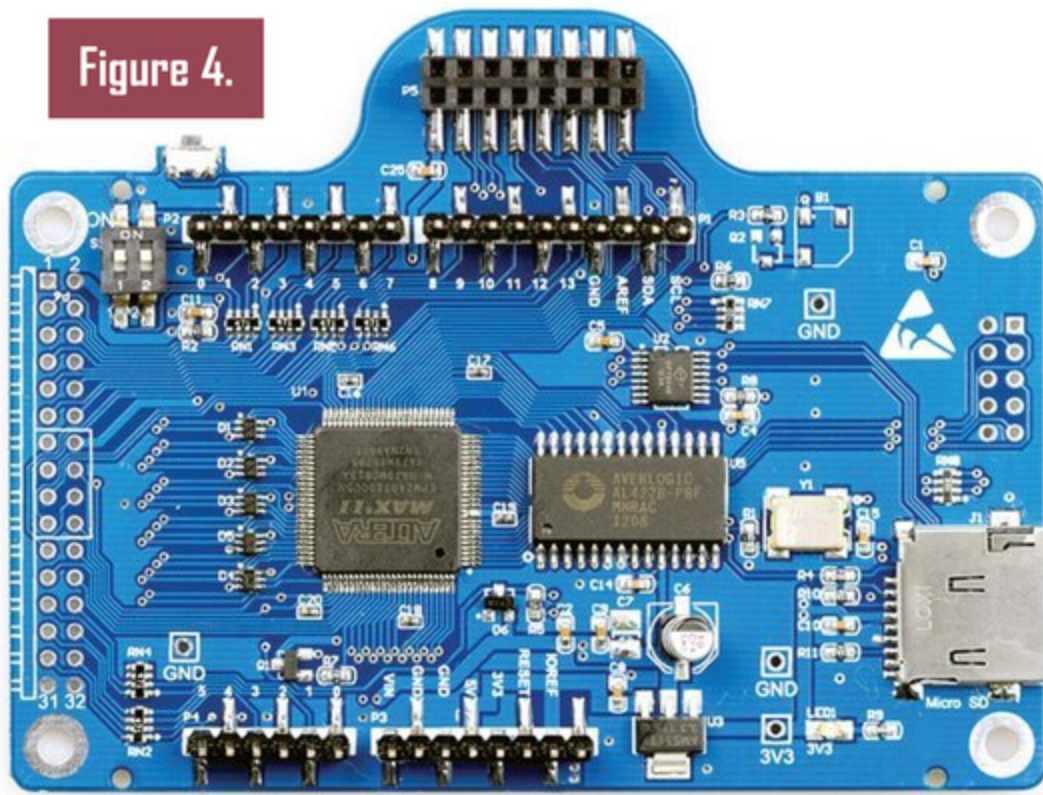


Figure 3.

4D Systems uCam-II

In my last two columns, I have been discussing the pros and cons of the uCam-II UART interface color camera from 4D Systems (see **Figure 2**). I got the camera communicating at nearly 1 Mbps to my controller board, which was about 3 fps. This isn't quick enough for most robot navigation, but can be used if the robot is not very fast. If you have a UART that can better match the uCam-II UART rates, then a bit rate of over 4 Mbps (12 fps?) could be possible, which is fast enough for navigation. This camera is \$49. 4D Systems

Figure 4.



There is one more entry into the Arducam suite: the Arducam Mini. This tiny board comes with chosen cameras installed on it and no uSD card slot. The interface is the same as the Arducam Shield I/O lines and the user guide is in the same place as well (www.arducam.com/category/user-guide/).

This device may not be as versatile as the Arducam Shield, but it is small enough that you can put more than one on your microcontroller board (see **Figure 5**). I have found these for sale at the UCTronics site, as well as in eBay stores for \$25.99 for a two megapixel camera model.

In Summary

If you want to “roll your own” vision system as a project, you can’t beat the tried and true Gameboy camera hack. It’ll be fun, educational, and inexpensive. This is the least expensive option that I can think of.

If you want a camera board that interfaces to an Arduino with example code, then the Arducam products are for you. You will have to roll your own vision software, but the hardware is not expensive. For a little more money and much better documentation, you can use the 4D Systems’ uCam-II camera and do your own vision algorithms. You will find code in the June 2015 Mr. Roboto column that will get you started. If you don’t mind spending a bit more money to get a ready-to-use vision system with object tracking capability, then the CMUcam5 is your best bet. This camera will get you “rolling” faster than any other vision system I’ve discussed.

No matter what you choose, you will learn something while giving your robot vision capabilities. Well, that is another several minutes you have used up reading Mr. Roboto! I hope that I have provided some food for thought when it comes to vision systems for your robots. Keep those questions coming to roboto@servomagazine.com and I’ll do my best to answer them! Until next time, keep building

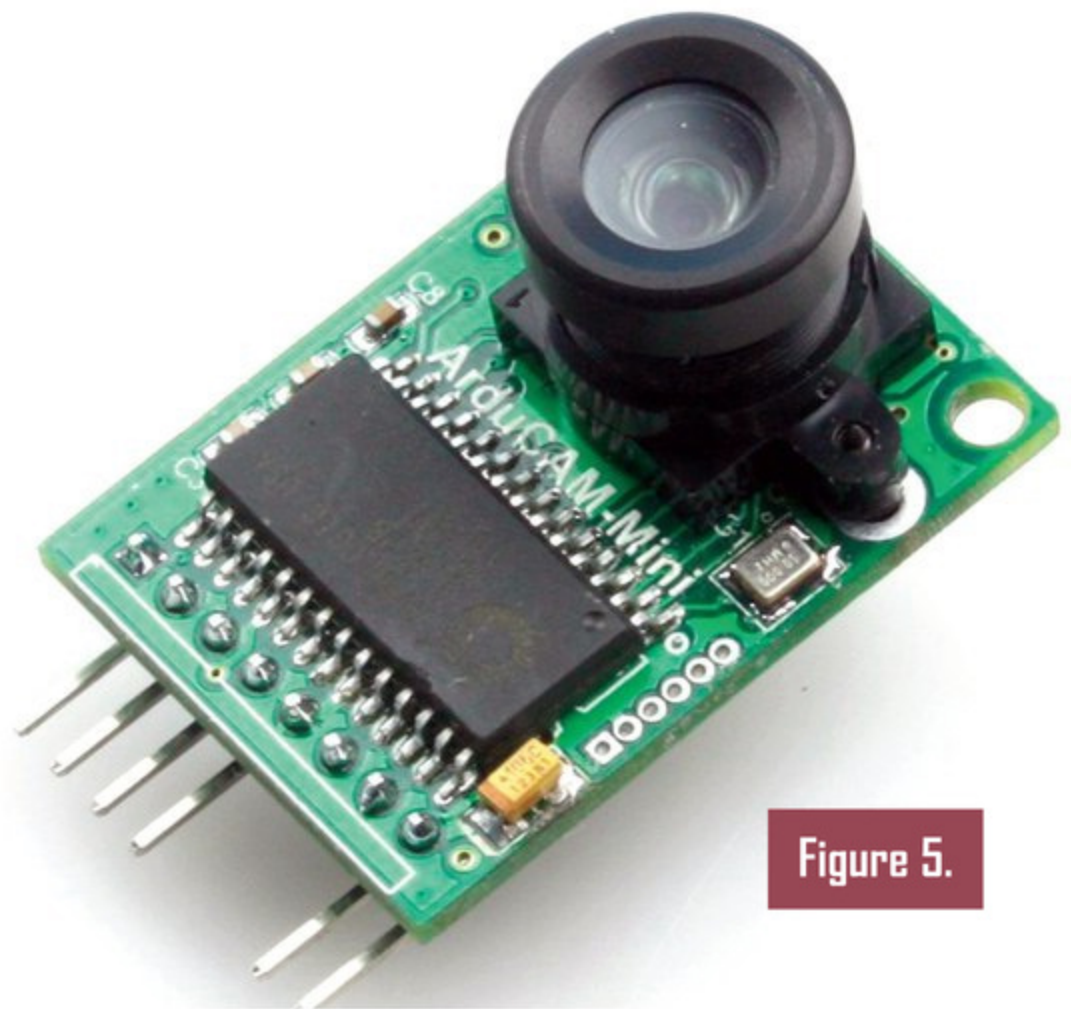


Figure 5.

robots. You want to be prepared to defend yourself with your own friendly robots when the evil robots descend from the sky to attack! **SV**

The Robot MarketPlace

Supporting Robotic Combat for over 10 years!

- Combat Robot Supplies
- Motors
- Batteries & Chargers
- Vex Robotics
- Carbon Fiber
- Wheels & Tires
- Speed Controllers
- R/C Systems
- Drive Components
- Hobby Supplies & Toys

(877) ROBOT 99
(877-762-6899)
or (941) 749-6030

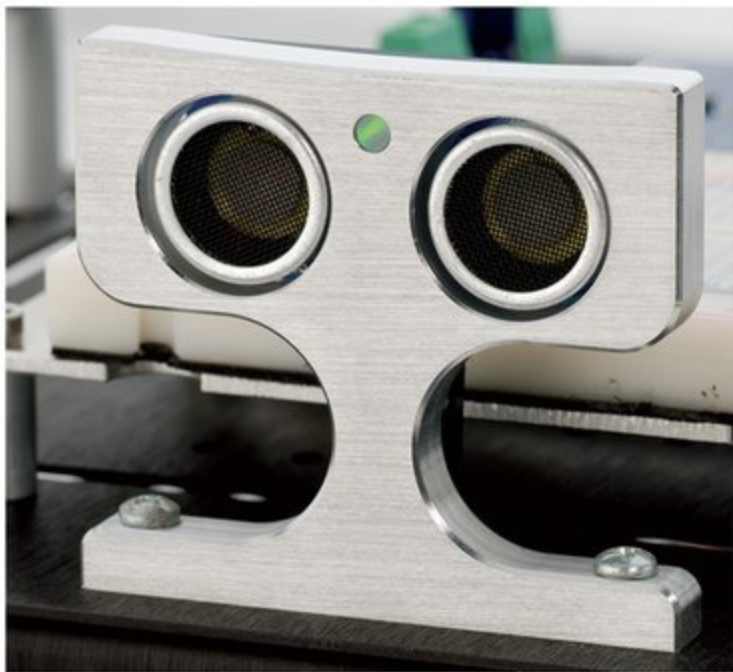
Your One-Stop Robot Shop!

RobotMarketPlace.com

NEW PRODUCTS

Machined Aluminum Sensor Stands

Parallax has recently added four new sensor stands to their current lineup. The PING))) Protector Stand (\$14.99) is a precisely machined mounting bracket cut from 1/4 in thick high-grade aluminum. With easy installation, this stand can attach the PING))) ultrasonic distance sensor to a robot and protect the sensor from most collisions.



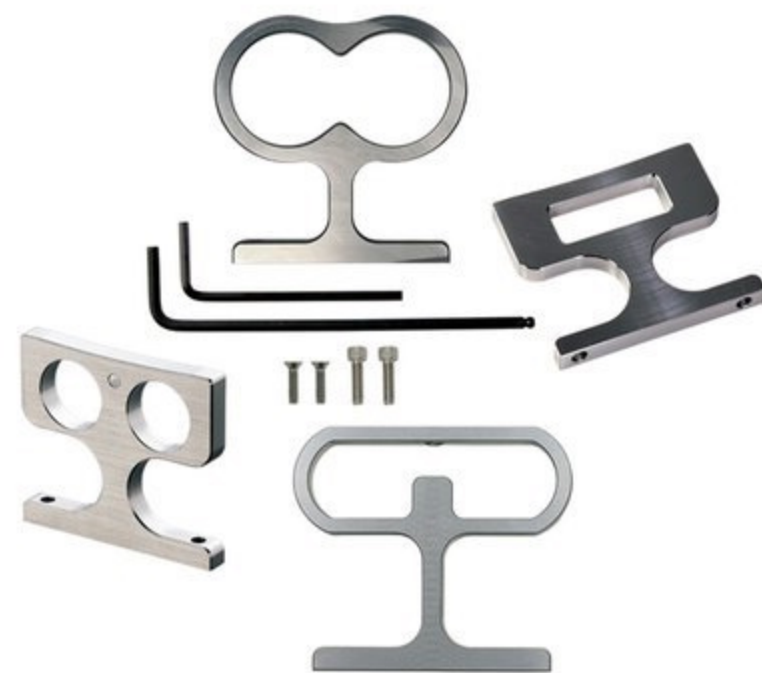
Key features include:

- Designed to protect the PING))) from damage.
- Hardware to connect the sensor to the stand is included.
- Custom CNC machined from 6061 high-grade aluminum.
- Aperture and plastic light pipe included to allow for visual check of sensor function while mounted to stand.

The next precision-machined bracket mount is for the Lightware SF10 series laser range finders (\$24.99). The stand is made from high-grade aluminum, and it conveniently positions and protects the laser range finder in both stationary and mobile applications. Assembly is quick and easy, with the necessary mounting hardware included.

Some features include:

- Hardware for mounting the sensor to the stand and then attaching the stand to a project is included.



- Mounting hole spacing is sized to fit pre-drilled holes in the Arlo robot platform.

- Dimensions (without sensor): 2.52 x 2.57 x 0.35 in (63.86 x 65.36 x 8.84 mm).

Parallax's Aluminum Stand Kit for Sharp IR distance sensors (\$12.99) is a precisely machined mounting bracket cut from 1/4 in thick high-grade aluminum. With easy installation, this stand can attach either of two popular Sharp IR distance sensors to a robot and protect the sensor from most collisions.

Hardware for mounting a sensor to the stand and attaching the stand to a project are included. This stand also has mounting holes that align with the pre-drilled holes in the Arlo robot base kit

Dimensions are: 1.95 x 2.20 x 0.25 in (11.43 x 55.88 x 6.35 mm).

The Aluminum Stand Kit (\$13.99) for Sharp GP2Y0A710K0Fs is a precision-machined mounting bracket milled from 3/8 in thick high-grade aluminum. Its clean design frames and protects the Sharp IR distance sensor. Mounting hardware and tools are included.

Some key features are:

- Secures and protects the Sharp GP2Y0A710K0F IR distance sensor from damage.

- 4-40 tapped mounting hole to

AndyMark
Robot Parts Experts

Thrifty Throttle

Run motor controllers or servos with AndyMark's easy and affordable PWM signal generator.



- PWM Generation
 - Standard servo range - 1000ms-2000ms
 - Arduino extended range - ~540ms-2300ms
- Comfortable one-hand operation
- 9V Battery Operation (battery not included)

Visit **AndyMark.com** and view our wide selection of robot parts!

Use coupon code **"SERVO5"** for 5% off your next order

Kokomo, Indiana • **www.AndyMark.com** • 877-868-4770

attach sensor to stand.

- Twin 6-32 thru-holes for mounting the sensor to a project.

- Mounting hole spacing designed to match the Arlo robot kit base.

- Stand dimensions (unassembled): 2.58 x 2.26 x 0.35 in (6.55 x 5.74 x .89 mm).

For further information, contact:



Parallax

www.parallax.com

CV2 Vision Systems with Faster Processing, More Cameras

Epson Robots introduces their new Epson CV2 Vision Systems. With speeds up to 8x faster than their CV1 vision systems, the CV2 versions provide tremendous processing power to help reduce cycle times and increase throughput.

Vision Guide has been used in factories throughout the world to allow their robots to "see." With a unique object-based point and click interface, Vision Guide allows developers to quickly create applications in hours vs. weeks or longer. Also included are the "vision to robot" calibration tools that take just minutes to set up for mobile and fixed camera calibration settings.

At the heart of Vision Guide is the unique object-based point and click interface which allows users to easily define ROIs and apply vision tools by simply dragging and dropping objects onto the image display area. Objects are strung together in a sequence to build a vision application quickly and easily. For example, a Geometric Search Tool could be used to find specific points of interest on a part and then line tools could be applied on top of the Geometric Search Tools to then



determine the distance between the two points of interest on a part.

Once the distance is checked and the part passes inspection, coordinates from the found location are automatically converted to robot coordinates and the robot moves to pick up the part using those coordinates, including X, Y position and rotation.

CV2 Vision Systems contain powerful CPUs and vision processing that runs in parallel with the Epson RC90 or RC700 controllers, allowing for maximum throughput. The vision systems are equipped with high flex cables (Ethernet or USB) and can be mounted to a robot arm (mobile) or mounted to a fixture (fixed). Wizard based calibration makes robot-to-vision system calibration simple, and allows for upward or downward camera calibrations. With the CV2 Vision Systems, both GigE and USB 2.0 cameras are supported. Up to six cameras (four GigE and two USB 2.0) can be connected to one system.

For further information, contact:

Epson Robots

www.epsonrobots.com

ALL ELECTRONICS CORPORATION

Thousands of Electronic Parts Available Today!

LEDs • CONNECTORS • RELAYS
SOLENOIDS • FANS • ENCLOSURES
MOTORS • WHEELS • MAGNETS
PC BOARDS • POWER SUPPLIES
SWITCHES • LIGHTS • BATTERIES
and many more items...

We have what you need for your next project.

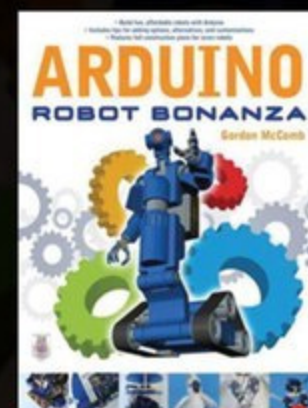


www.allelectronics.com

Discount Prices • Fast Shipping

SERVO

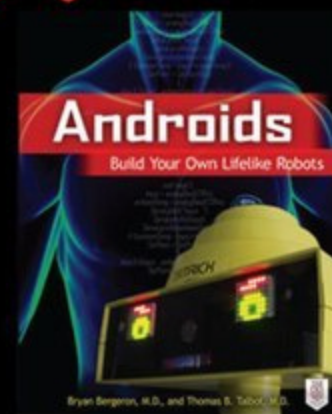
SERVO BOOK SPECIALS



413 Pages
Only \$23.95

SAVE 20%!

336 Pages
Only \$23.95



To order call 800 783-4624 or visit:
<http://store@servomagazine.com>

Limited time offer

bots IN BRIEF

DRONE TO THE RESCUE

Recent powerful storms dumped heavy rain on parts of North Texas, causing power outages and flash flooding throughout the area.

Johnson County alone reported 13 water rescues, and one of those came via drone.

The Kastel family lives in a mobile home that was completely surrounded by water. The fast-moving water made it impossible for firefighters to reach the family, according to Fox 4 News.

That's when Johnson County emergency personnel decided to use a DJI drone to get a rescue line to the Kastels. Garrett Bryl has volunteered his drone called "Valkyrie" for months to help in emergency situations.

Johnson County Emergency Management Director, Jamie Moore said, "They attached a small lead line; in other words, a very small rope to the bottom of the drone so that rescuers had a rope between them and the house to be able to pull themselves across the river."

Drones are being used more and more to aid in disaster relief. Nepal — which suffered its worst earthquake in 80 years on April 25, 2015 — used drones for aerial photography and to map areas affected by the earthquake. That information is passed on to aid crews and rescue workers on the ground.

Emergency personnel in Johnson County, TX attempt to rescue a family from a flooded home. (Photo courtesy of Gary Ultee/WFAA.)



FOLDS UNDER PRESSURE

At the recent International Conference on Robotics and Automation, MIT researchers presented a printable Origami robot that folds itself up from a flat sheet of plastic when heated, and measures about a centimeter from front to back.

Weighing only a third of a gram, the robot can swim, climb an incline, traverse rough terrain, and carry a load twice its weight. Other than the self-folding plastic sheet, the robot's only component is a permanent magnet affixed to its back. Its motions are controlled by external magnetic fields.

Once the robot has folded itself up, the proper application of a magnetic field to the permanent magnet on its back causes its body to flex. The friction between the robot's front feet and the ground is great enough that the front feet stay fixed while the back feet lift. Then, another sequence of magnetic fields causes the robot's body to twist slightly, which breaks the front feet's adhesion, and the robot moves forward.

MIT's untethered miniature Origami robot could have a variety of medical uses when introduced inside of a human body, potentially zapping cancer cells or unclogging arteries.



GOOGLE'S DRIVING RECORD

A recent report shed some light on the number of accidents Google's self-driving cars have been involved in.

Google says its group of 20 autonomous cars have been involved in 11 accidents since hitting the roads in 2009. All the accidents — which were minor and did not cause injury — were actually caused by humans driving other cars. Google stated that none of the accidents were caused by its self-driving cars.

Here's a breakdown of the accidents:

- Seven accidents involved Google cars being rear-ended
- Two were side-swipes
- One accident involved a car rolling through a stop sign
- Eight of the accidents occurred on city streets



WE BE JAMMIN'

Soft actuators are appealing for robotics because they're cheap (made out of plastics or polymers and air), inherently compliant, and relatively safe for humans to interact with. Plus, they are able to adapt themselves to grip a wide range of objects. Being soft is fine and dandy, unless you need an actuator with some stiffness. So, what do you do?

Well, researchers at Technische Universitat Berlin led by Professor Oliver Brock have combined soft pneumatic actuators with a jamming system that results in a variable-stiffness actuator that's soft when you want or hard when you want. They call it Pneuflex.

The researchers tested three different jamming systems, including the traditional coffee grounds, as well as two other designs based on scales and interleaved layers.

They eventually settled on the interleaved layers for the final design because it requires far less pressure to jam — although it's more complex to manufacture.

In principle, it works in a similar manner: When the interleaved layers have air between them, they can slide against each other, allowing the actuator to flex. When the air is pumped out, the layers compress against each other, and the actuator is stiffened. You can do this by hand with a syringe, or using a pump or autonomous operation.

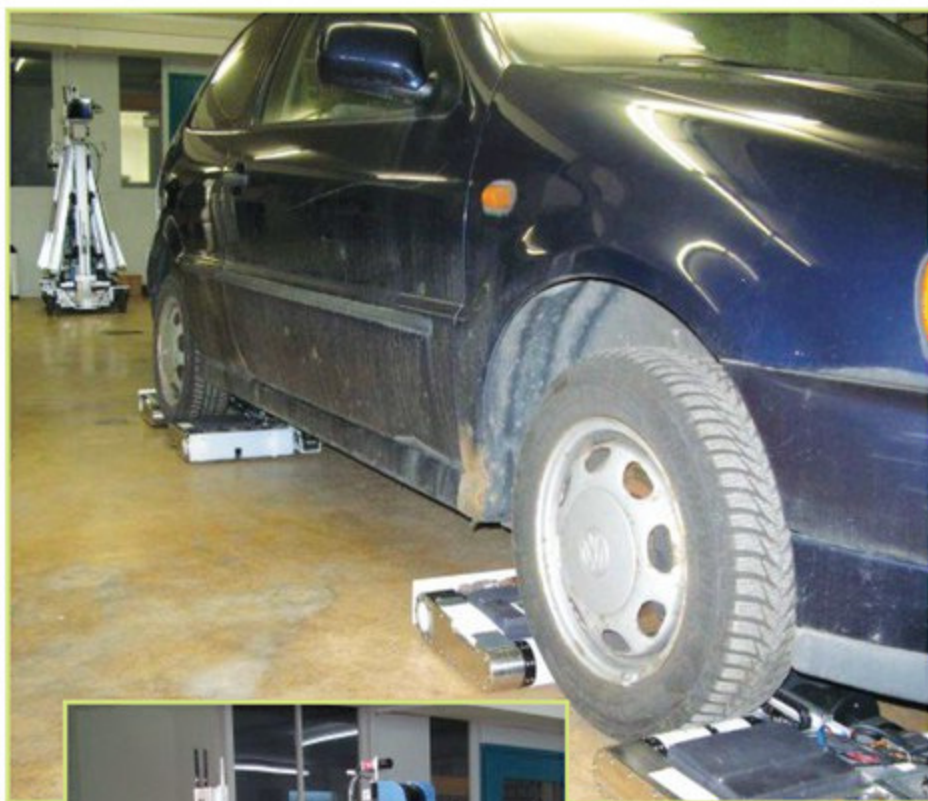
The three jamming systems tested included: granular jamming; layer jamming with overlapping fish-scale-like layers; and layer jamming with stacks of interleaved layers.

Overall, the jammed actuator exhibited a stiffness increase of 8x, resulting in an application of force increase of 2.3x — which is pretty significant.

Incidentally, you can make these actuators yourself for free. Instructions are available at www.robotics.tu-berlin.de/menue/research/compliant_manipulators/pneuflex_tutorial/.



Photo: Evan Ackerman
IEEE Spectrum



TOWING AVERTed

AVERT — which stands for Autonomous Multi-Robot System for Vehicle Extraction and Transportation — is made up of four tiny robots and a larger deployment unit that lifts vehicles up to two tons. Fitted with two lasers and a digital camera, the robot scans the area looking for possible obstacles and plans the safest way to remove them.

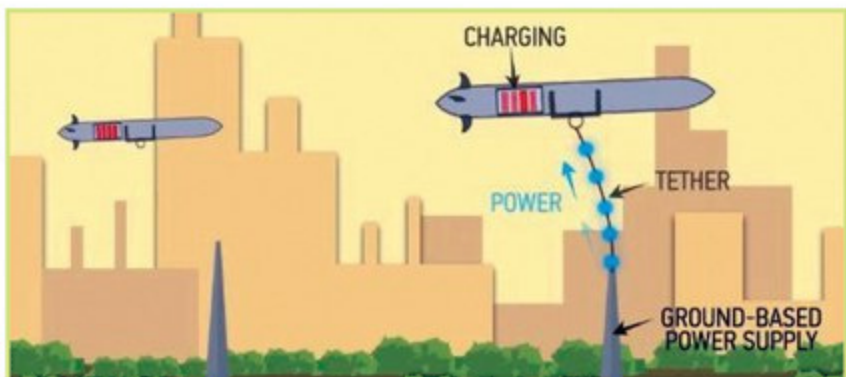
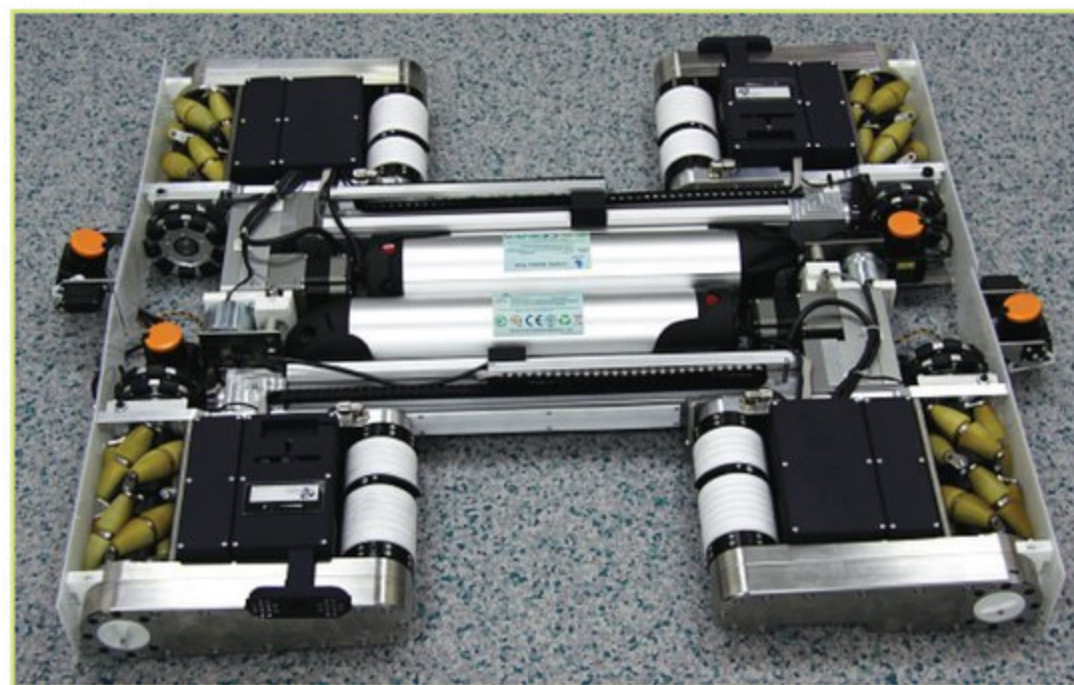
The small robots are then deployed and attach themselves to the wheels of the vehicle that needs extraction. It only lifts the car an inch off the ground, but that's enough for the tiny bots to safely move the car.

The device was specifically developed to be used by law enforcement. AVERT can be used to remove suspicious or obstructing vehicles from enclosed spaces like buildings, tunnels, underground car garages, and low bridges. The tiny robots were designed to remove vehicles from tight enclosures where other tow vehicles can't go.

Since AVERT has delicate handling, there's no need to worry about the vehicles getting damaged. In addition, these bots present a safe way

to remove cars that may be carrying explosives or other hazardous materials without human intervention.

The system has been in development since 2012, and production is expected to start sometime in 2016.



MID-AIR HOOKUPS

Battery life is an issue when it comes to drones. Whether it's a consumer drone, commercial drone, or military drone, everyone would like these flying bots to be able to stay in the air for longer.

Boeing might have found a way to do just that. In fact, they were granted a patent that works only for airship-style and other buoyant drones.

The patent describes a system that uses tether-equipped drones that can connect with ground-based power supplies to recharge them while airborne. When the drone is completely charged, it continues on its merry way making room for another drone to use the recharging station.

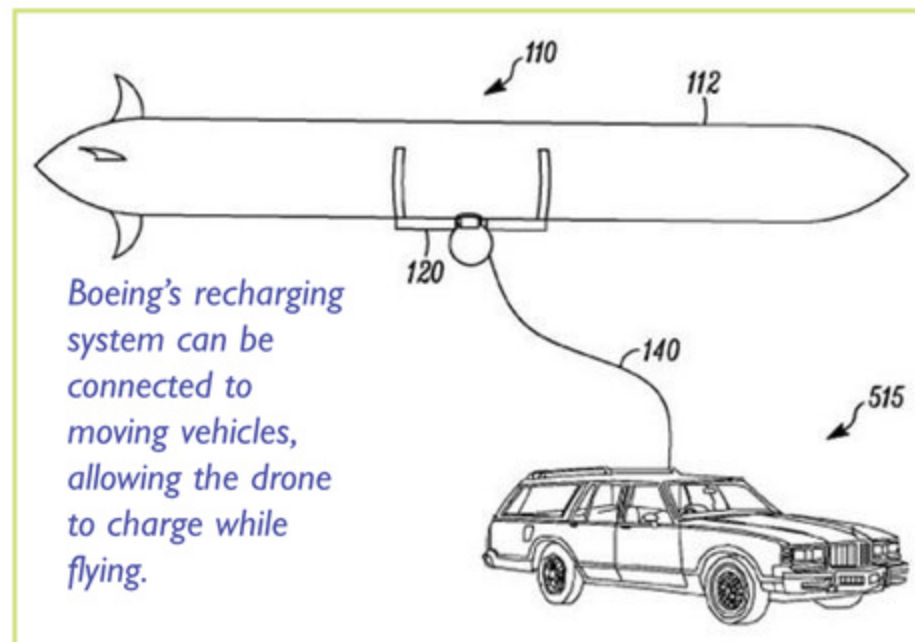
According to the patent, drones would either come equipped with a tether that connects to the charging station, or they'd fly up to a floating tether and connect that way. The tether can even be connected to moving vehicles, allowing the drone to charge while flying.

Again, this patent is designed for airship-style drones, so it might not yet be applicable for the more popular

quadcopter-style consumer drone. It's just the latest attempt to improve flying time.

A Singapore-based company called Horizon Unmanned Systems recently unveiled its Hycopter drone that runs on hydrogen power and can fly for four hours at a time — 2.5 hours when carrying a 2.2 pound payload.

Hycopter uses its frame to store energy in the form of hydrogen instead of air, eliminating energy storage weight.



AUGUST

6-8 Defconbots
Las Vegas, NV
Autonomous Tower Defense.
www.defconbots.org

12-14 AUVS International Aerial Robotics Competition
Atlanta, GA and
Beijing, China
Fully autonomous air vehicle and sub-vehicles perform a series of tasks.
www.aerialroboticscompetition.org

16 Missouri State Fair Robot Expo
Sedalia, MO
Events include FIRST LEGO League, FIRST Tech Challenge, FIRST Robotics Competition, VEX Robotics Competition, 4-H Robot Challenge, and other contests.
www.mostatefair.com

31 World Robotic Sailing Championships
Åland Islands, Finland

Know of any robot competitions I've missed? Is your local school or robot group planning a contest? Send an email to steve@ncc.com and tell me about it. Be sure to include the date and location of your contest. If you have a website with contest info, send along the URL as well, so we can tell everyone else about it. For last-minute updates and changes, you can always find the most recent version of the Robot Competition FAQ at Robots.net: <http://robots.net/rcfaq.html>.

— R. Steven Rainwater

Autonomous robot sailboats must navigate an ocean course around buoys. Contest runs through September 4.
www.roboticsailing.org

SEPTEMBER

4-7 DragonCon Robot Battles
Atlanta, GA
Autonomous and remote control robot battles in various weight ranges.
www.dragoncon.org



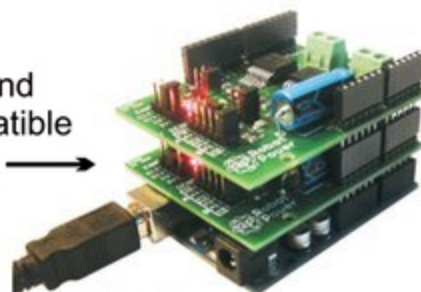
www.robotpower.com

MEGAMOTO GT Shield



New

Software and plug compatible with our MegaMoto shield



Our Highest Current motor control for Arduino™

- ◆ 35A Continuous 50A peak
- ◆ 6.5V 40V
- ◆ Built in Voltage Reg can power the Arduino
- ◆ Current sensor outputs can be sent to any analog pin
- ◆ Over temp and over current protection



The Hydra

- ◆ 7V 28V
- ◆ Triple 6.5A / 8.0A Peak H Bridges
- ◆ Current/Temp limiting
- ◆ R/C inputs w/mixing
- ◆ Limit switches on one channel
- ◆ 2.0" x 2.1" x .5"



Big Bots!
Big Power!
Big Motors!



The Vyper

- ◆ 120A+ H-Bridge - 200A Peak!
- ◆ 8V 42V
- ◆ Overload protected
- ◆ RC and Analog Pot input
- ◆ Supports 8 AWG wires!
- ◆ 2.85" x 2.25" x 1.75"

Phone: 360-515-0691 • sales@robotpower.com

THE COOLEST CLASSROOM EVER

Educators get 15% off their entire order*



PARALLAX

www.parallax.com

*Full program details: www.parallax.com/teach

COMBAT ZONE

Post comments on this section and find any associated files and/or downloads at www.servomagazine.com/index.php/magazine/article/august2015_CombatZone.

Featured This Month:

- 16** *Back to Do Battle*
by **Andrea Suarez**
- 18** *Building Better Bots: The Weak Link*
by **Russ Barrow**
- 20** *BUILD REPORT: The Build of Ash and Its First Event*
by **Jack Tweedy**
- 23** *EVENT REPORT: Freeside Robot Street Fight 2015*
by **Michael Jeffries**
- 25** *Cartoon*

Back to Do Battle

● by Andrea Suarez

If you're reading this magazine, you've probably spent hours watching BattleBots episodes full of mayhem and destruction. You've seen Nightmare, Son of Whyachi, and Diesector rule the arena with their spectacular performances. For us, the builders of these machines became the heroes of the sport. Jim Smentowski, Terry Ewert, and Donald

Hutson (creators of the robots mentioned earlier) were our celebrities.

If we had a chance to meet them, we were more awestruck than if we were meeting any rock band or sports team. They were always excited to meet the new builders and willing to indulge our questions and encourage our participation. They were superstars.



Mike Gellatly (Witch Doctor) and Jim Smentowski (Nightmare) fight robots in the hotel lobby. Thanks Bot Bash!



Shaman competed with all these team autographs.

As soon as we found out we were selected to compete in the new ABC BattleBots series, our thoughts turned to these BattleBots veterans. Do you think Warhead will be there? How about Phrizbee? Suddenly, we had to build a machine that would be competitive against the legends of the sport, and we only had a few short weeks to do it.

What made this build even more daunting was that we had no idea what any of our competitors were building. We were all legally bound to keep our designs a secret. We could only imagine what these builders would conjure up!

As we first arrived at the venue, my attention was instantly drawn to the large crates with spray-painted wording. Lockjaw, Captain Shrederator, Stinger, Bronco! This is it — this is the real BattleBots!

Before the fighting even started, I walked around with our robot's top plate and got it signed by every team I could find. Everyone was so eager to get to know each other, it took hours to make my way through the pits.

We were all about to face-off in the most intense competition in the history of robot combat. It was exhilarating and it was terrifying. It made us question every design

decision, and visualize every worst-case scenario.

Our robot, Witch Doctor, was introduced as the "newbie" time and time again. Of course, we felt we were far from newbies. We had been competing at the middleweight (120 lb) level for over 10 years and we had won several national events. We knew the names of almost every team there, but almost no one *knew us*.

We realized this gave us a few advantages. While many veteran teams were coming off of a 10 year hiatus for this event, our team spent the last decade honing our skills at over 30 events across the country. In that time, new technology has allowed for stronger, more powerful machines.

Lithium polymer batteries have significantly increased the current delivery a robot can exploit. Brushless motors can deliver much more power than brushed motors, often in less than a third of the weight.

This new level of technology takes experience to master, and we were hoping our veteran competitors were still using the components they trusted in the original series.

The now late Buzz Dawson and Team Da Vinci are known for their



Go Witch Doctor! Photo by Daniel Longmire.

unforgettable entrances into the arena with Moebius. The team would dress up as monks and raise Moebius out of their full-sized coffin filled with green fog. While we couldn't rival this performance, we knew we wanted our team to stand out.

We set out to complete our look

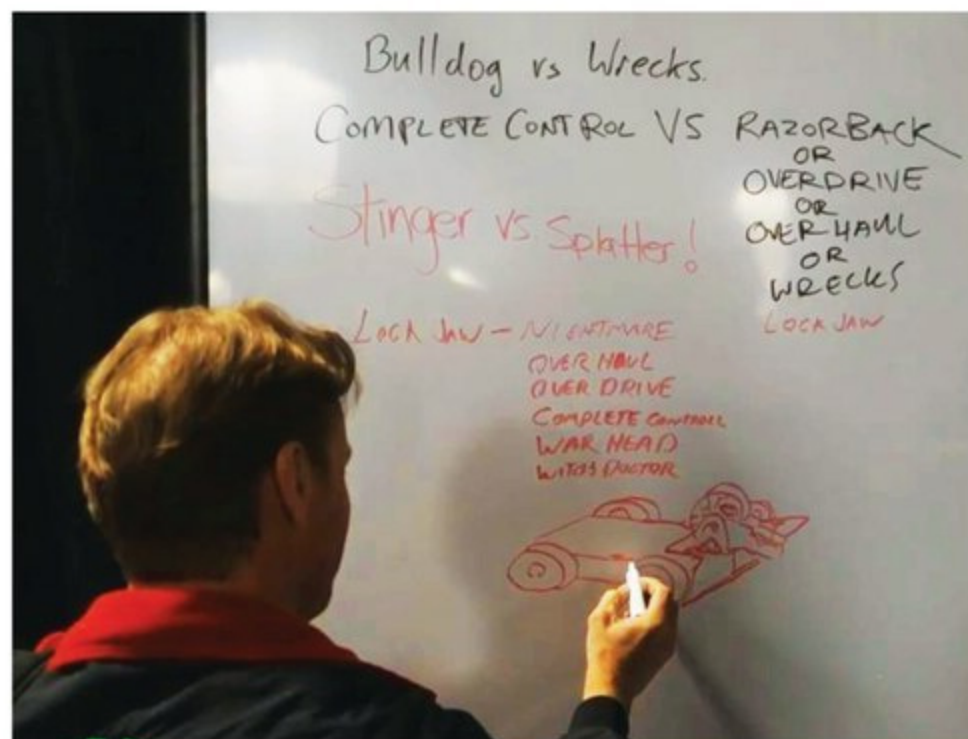


Buzz Dawson lifts Moebius from the dead!

Photo by Michael "Fuzzy" Mauldin.



Safety inspections in costume.



Donald Hutson (Mutant Robots, Diesector) draws Witch Doctor as he challenges us to a grudge match.

with voodoo-themed top hats lined with candles, feathers, and skulls.

As Buzz wrote on his site: "Now, the question was, with all this showmanship would [the robot] perform?" We were the "new" team, and we were in full voodoo garb. If they didn't notice us before, everyone was watching now.

Once the first round fights were announced, the playing field was suddenly leveled. Our opponents came to our pit table to examine Witch Doctor and Shaman, and they seemed as intimidated as we were. Inertia Labs (prior BattleBots

Champions) took Bronco off to their nearby shop as soon as the fight schedule was announced.

We thought this was just an intimidation tactic, but they finally returned just hours before our fight with lowered armor and an extended flipping arm.

The Lincoln Electric welding station often had a waiting line of competitors anxious to modify their robots before their next match.

Even for our match against one of the most formidable modern heavyweight robots — Tombstone of Hardcore Robotics — switched to their

shorter, heavier bar to make the notorious bar spinner more stable and controllable. This was the long-awaited return of BattleBots, and no one was taking any chances.

At a typical open-entry event, you meet some experienced teams and some first-time builders, but most teams fall in the middle of the pack. So, what happens when the competitors are hand-picked?

You get the most difficult, most rewarding experience a robot builder can ever imagine! I can't wait to meet next year's "new" teams! **SV**

Building Better Bots: The Weak Link

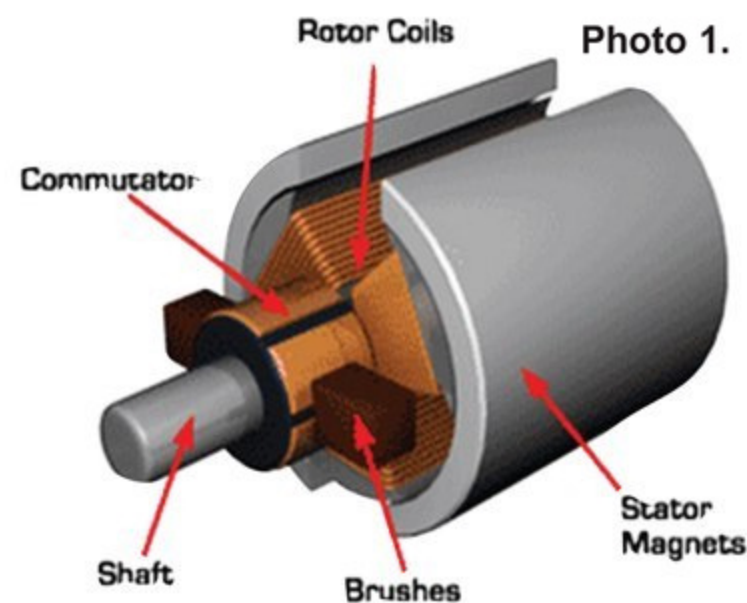
● by Russ Barrow

In building any robotic system, every part of the system must be tuned to provide the maximum possible reliability and performance. Simple things like loose screws, exposed wires, or fatigued batteries can turn even the most brilliant design into a smoldering mess.

One component — often neglected or assumed reliable — may be waiting to fail at the worst possible

minute. That brushed motor you rely on to keep your robot moving could be your weak link. DC brushed motors are the core of most mobile robot's drive-train. They are simple to use; you just apply power and go.

This ease of use is based on small carbon/metal brushes, spring-loaded against commutator bars (segmented arc shaped copper arranged in a circle) on the motor commutator



(center rotating assembly of a brushed motor). **Photo 1** shows a basic DC brushed motor.

When power is applied to the motor brushes, they transfer electricity through the commutator bars and energize a section of wire that generates a magnetic field. Stationary permanent magnets arranged around the commutator produce another magnetic field that repel and align with the commutator, causing it to rotate.

As the motor rotates, the next segment of commutator bars and coils are energized, and the process repeats as long as power is applied. The contact of the brushes to the commutator is also the weakness of this motor type.

The most obvious problem is wear of the brushes and the surface of the commutator bars. The commutator bars each have a gap between them, so the brushes get worn by the gap but also experience an electrical arc when the power transitions from one group of bar and coils to the next.

This electrical arc vaporizes or melts a small amount of brush and bar material. As this happens, the surface gets rougher and the wear accelerates from the bouncing of the brushes and even more arcing.

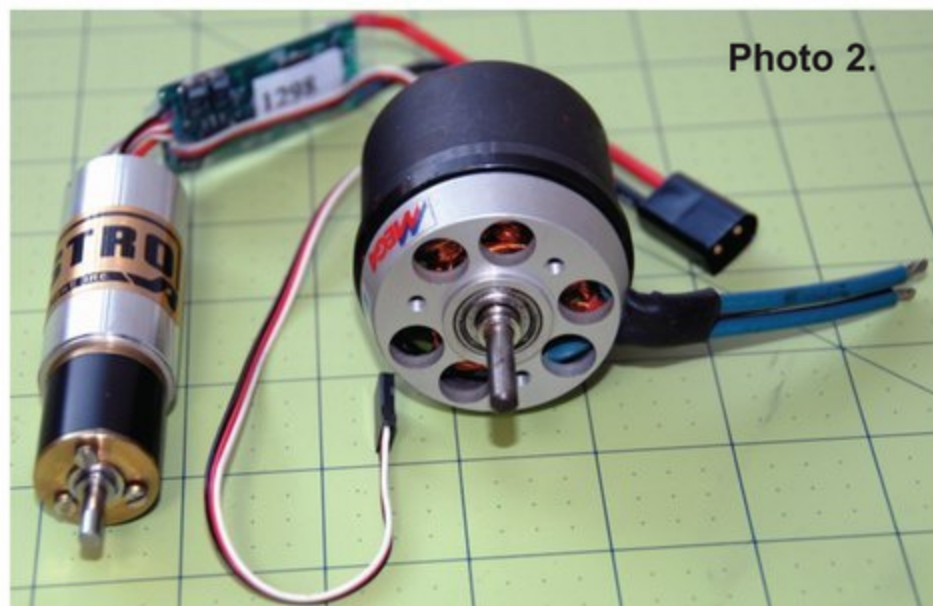
A more significant problem with the brushed motor is the fragileness of the spring supported brush and the thermally isolated armature. If, for example, a shock load caused the brush to fracture or break off, a rough edge of the remaining brush might dig into the space between the commutator bars and continue to fragment. This shrapnel will then likely jam the motor during rotation, causing the motor to short out and draw large amounts of current.

Another possible problem is that the rapid brush wear can cause large amounts of conductive dust to deposit between the commutator bars, shorting them together. This will result in multiple winds being energized, less efficient motor operation, and significant heat on the armature coils.

Unlike wear — where the motor eventually operates sporadically — damage of the brushes or commutator bars can destroy controllers, batteries, and the motor by burning the insulation on the motor coils.

There is an alternative to the brushed motor. Brushless motors are becoming more common due to low cost controllers or ESCs (electronic speed controllers), and a shift to this technology. Unlike brushed motors — which are mechanically commutated — brushless motors are controlled solely by the application of power through three individual coils that are wound around a stator with a rotating rotor inside the stator (called an inrunner), or with a rotating can shaped cylinder outside the stator (called an outrunner).

Photo 2 shows an inrunner on the left and an outrunner on the right. Although mechanically the two



motor types appear very different, both brushless motors are electrically commutated in the same way.

Brushless motor control is not simple, and there are people with PhDs who specialize in control. Fortunately for us, it can be boiled down into three control types: servo, sensed, and sensorless.

Servo based control relies on an encoder to provide the position, velocity, and acceleration of the rotor back to the controller. This allows for some very complex math that can then be used to accurately control the speed and torque, or precisely move the rotor to any angular position.

The most basic sensed control uses at least three Hall-effect sensors (non-contact sensors that detect magnetic fields) to roughly determine the rotor's magnet position and alter the frequency of the power applied.

Sensorless control works by energizing one of the three coils (randomly), then detecting a change in the magnetic field through the other two windings which is read by the controller as voltage. This is often referred to as back electromotive force (back EMF). This defines rotation direction and allows the controller to then energize the coils in a sequence that achieves the desired rotation direction and speed.

For most robotic applications — especially for wheeled locomotion — we are mainly looking at sensed or sensorless control. Sensed brushless motors can produce higher torque than sensorless brushless motors, as the Hall sensors provide feedback on

how the rotor is responding to the load. Therefore, the controller can change frequency and power to match inertia and torque requirements.

Sensorless controllers will have lower torques than both a brushed motor and a sensed or servo controlled brushless, since they have very limited rotor positioning feedback. High inertia loads can cause the rotor to lag the magnetic field

generated through the coils as driven by the controller, failing to start rotation or lose efficiency during startup. **Photo 3** shows a common



sensored inrunner brushless motor.

For small weight sensitive robotic applications, motor dimensions and controller weight might dictate a sensorless control. Since startup torque is limited and the potential for brief motor reversal is possible, the motor must be matched to the load. Typically, this requires a method to either decouple the motor from the load during startup, or enough torque multiplication through a gearbox or other reduction method.

To drive locomotion wheels, fine control of movement will require even additional reduction. This is because both sensorless and sensed brushless motors lack smoothness below 100-200 RPM.

On my combat robot, Dark Pummerler, I used an outrunner style sensorless brushless motor with a reversible sensorless brushless controller. The motor has a maximum speed of 27,000 RPM which can easily be handled by the controller, but the lowest continuous speed I have been able to measure is around 3,000 RPM.

Below that speed, the motor tends to reverse, twitch, and then spin

up and down.

This is a factor of the controller trying to execute a startup rotation sequence, along with factors like a radio control deadband (center stick off region), back EMF detection, and minimal torque. Originally, I worked with a low total reduction to a small diameter wheel. The robot had good mid/full speed torque — easily capable of spinning the wheels — and was almost uncontrollably fast. However,

this happened with a brush motor and controller, I would have permanently damaged the controller and the motor — potentially even the batteries as a stalled brushed motor would have consumed a massive amount of current.

With the brushless motor and controller, I simply fixed the issue and kept competing. I will continue to refine the design and try to remove any more weak links. **SV**

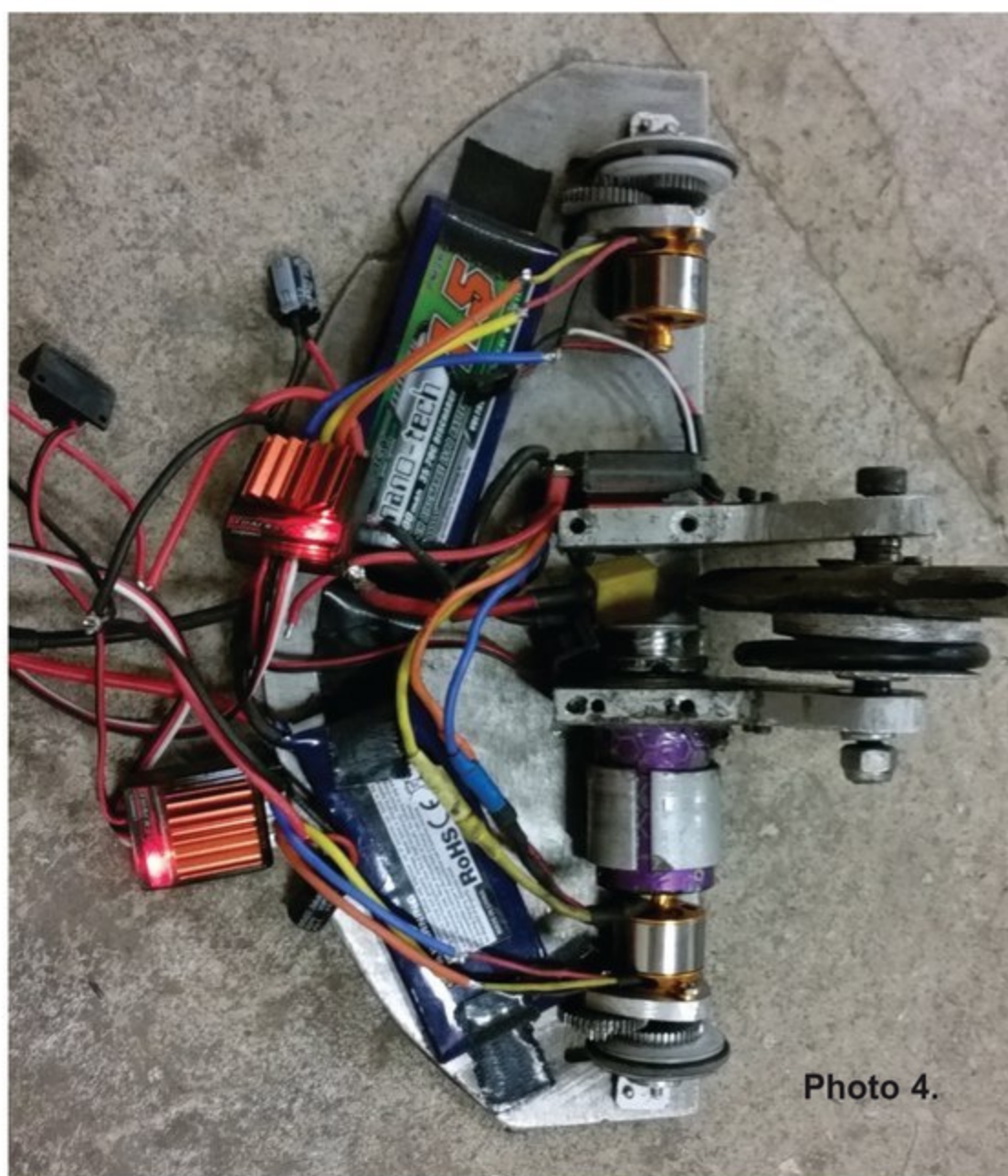


Photo 4.

low speed maneuvering was poor.

The robot would often stray off center on a forward command and oversteer on a turn. So, while trying to solve a reliability problem, I created a control issue. All is not lost, however. I can easily trade some top speed for better fine control, so a bit more gearbox reduction should solve both issues.

Photo 4 shows the armor off for drive testing.

Recently, I competed with this robot, and had a variety of problems in addition to control issues. This included O-rings mounted to plastic hubs for wheels for traction. In two matches, both O-rings on the two wheels came off and bound up the wheel. Had

BUILD REPORT: The Build of Ash and Its First Event

● by Jack Tweedy

Ash was intended initially to be an improved version of the UK Beetleweight, RPM, although somewhere through the design process, my friend (the infinitely better than myself, Harry Hills) suggested I go for a four wheel drive vertical

spinner. Thus, Ash was born in concept. It all started with a CAD drawing that Harry had (**Photo 1**).

After this had been finalized, the parts went off for CNC milling and were eventually sent to Harry for some pocketing (**Photo 2**).

At the time Harry received the parts for Ash, he was also preparing for the Featherweight World Champs with Massacre, his overhead bar spinner. That left me with two days to get Ash going after it arrived the Thursday before the Robots Live

Burgess Hill event.

Generously, Harry made a couple of HDPE lids, an HDPE baseplate, and front plates to both save weight and reduce how much time it would take me to get Ash ready.

After Ash had arrived, a wedge needed to be made, wiring needed to be done, and some modifications to the baseplate were also necessary. First of all, I had to countersink the bolts holding the baseplate on as the button head bolts that were being used were preventing the wheels from touching the floor.

After this, the wedge was needed. I had a couple of small four millimeter steel plates lying around; it was complete overkill for a bot in such a light class, but it was what I had on hand and it worked (**Photo 3**).

So, Ash was ready to go and the minimal amount of testing I got to do seemed promising. Little did I know that the event would be plagued by Beetleweight issues on all except for two bots.

Both Ash and Terminus (my horizontal spinner) blew receivers because for some reason the BECs that I was using on both of them were causing the receivers to — seemingly at random — take the last input given and run with it for a good five seconds before returning control.

After borrowing a receiver and

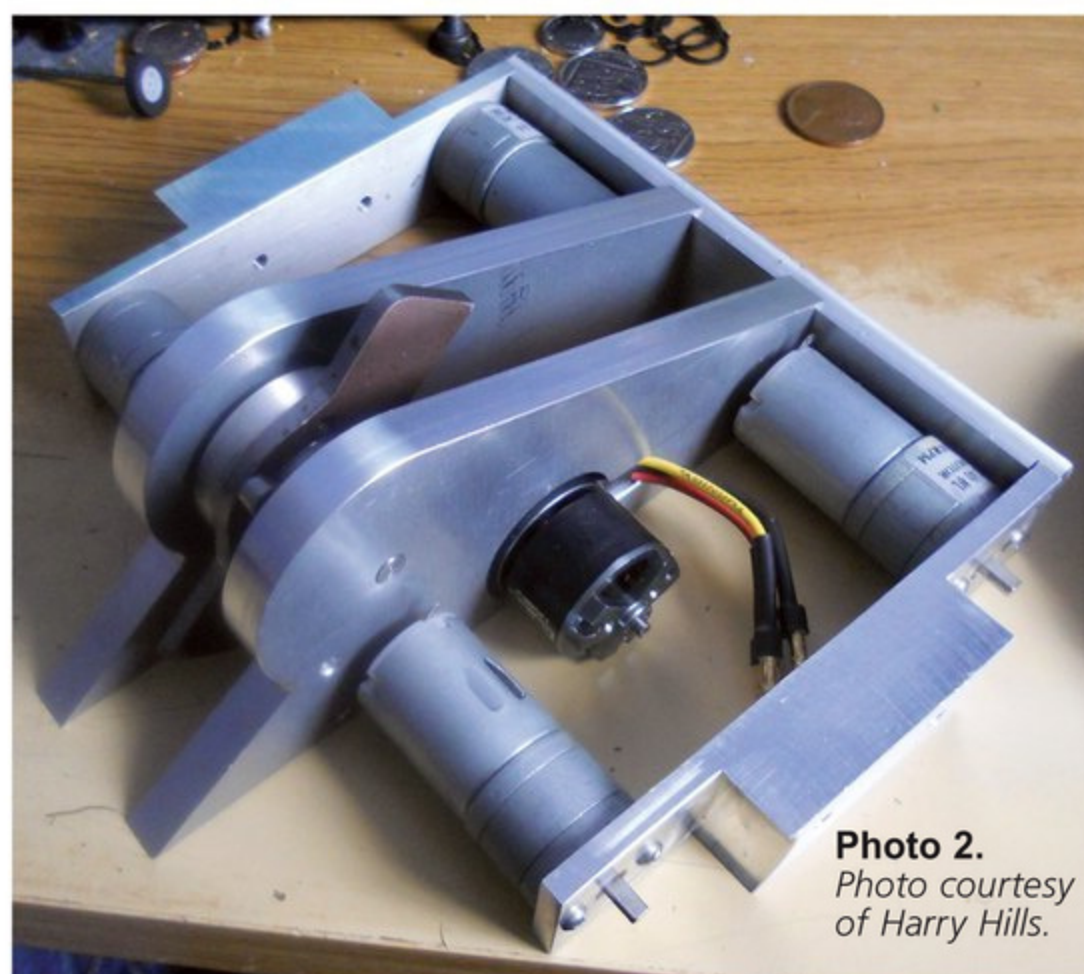


Photo 2.
Photo courtesy
of Harry Hills.

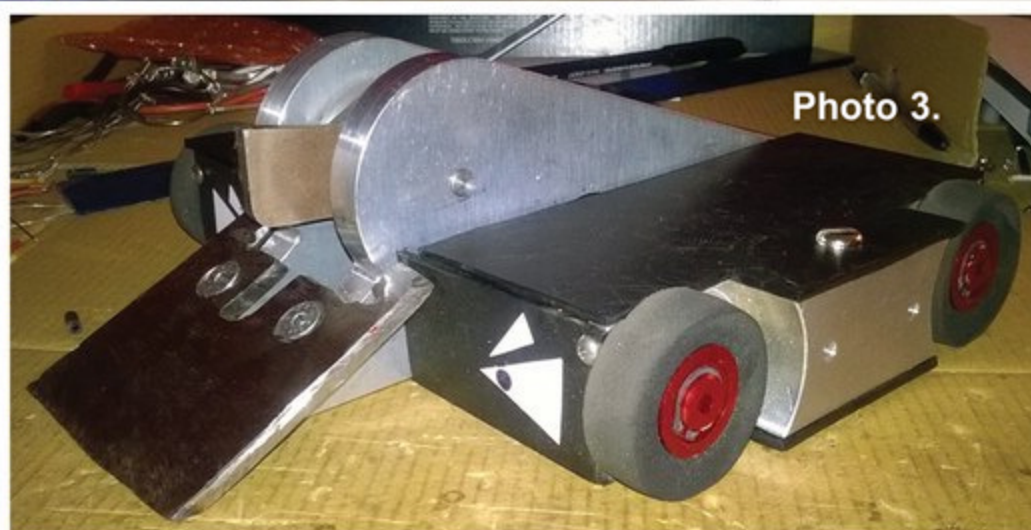


Photo 3.

receiver pack from David Weston, Ash ran as intended. However, the size of the receiver pack would mean that the lid on one side wouldn't fit, so a new one was crafted from tape and hope.

Luckily enough for my bot's sake, the only other spinner there (RPM) also had issues, blowing one H-bridge on an ESC and losing reverse on its left wheel before blowing a brushless motor seemingly out of the blue during a demo.

This left Ash as the only functioning spinner there. Fast forward to Day 2 of the event, and we had arranged a fight between Ash and my friend, Craig's Beetleweight lifter, Mr. Snappy. This was to be Ash's first actual fight against anything other than a few empty cans to test the spinner.

It started off with a lot

of trying to drive around the incredibly wide opponent as the acetate I had attached to the wedge on Ash became more of a hinderance than a help. A few glancing blows hit, and then Ash's receiver pack ran out leaving Mr. Snappy as the victor.

Fight two was very similar — a few glancing blows with one decent hit sent Snappy into the arena wall and bouncing back onto its wheels. The fight ended when — while trying to manouver around the tiny frame of Ash — Snappy drove head first into the pit giving Ash the win.

Fight three, however, was where Ash shone as a bot. The fight started immediately with the bots going wedge to wedge. Ash came up the winner, forcing Snappy to run directly into the weapon. However, due to my box rush tactic, the weapon had not spun up to speed and instead of throwing Snappy, Ash simply nudged it up onto the top of the frame. That did allow me to push Snappy around.

After letting Snappy down, Ash's weapon ground up against the hinged wedge of Snappy a few times before getting under Snappy with the weapon spun up and landing a decent hit, sending Snappy toppling over backwards.

However, my eagerness to gain control of the fight meant I had driven too far forward, and hit the bottom of Snappy as it was flipping over, flipping it back onto its baseplate. After this, Snappy was now on the defensive and tried to make an escape to the opposite end of the arena. Ash pursued with its higher speed and forced Snappy towards the pit, making the robot either go straight into the pit or reverse directly onto Ash's weapon. The latter was chosen, and Ash got a nice hit on the

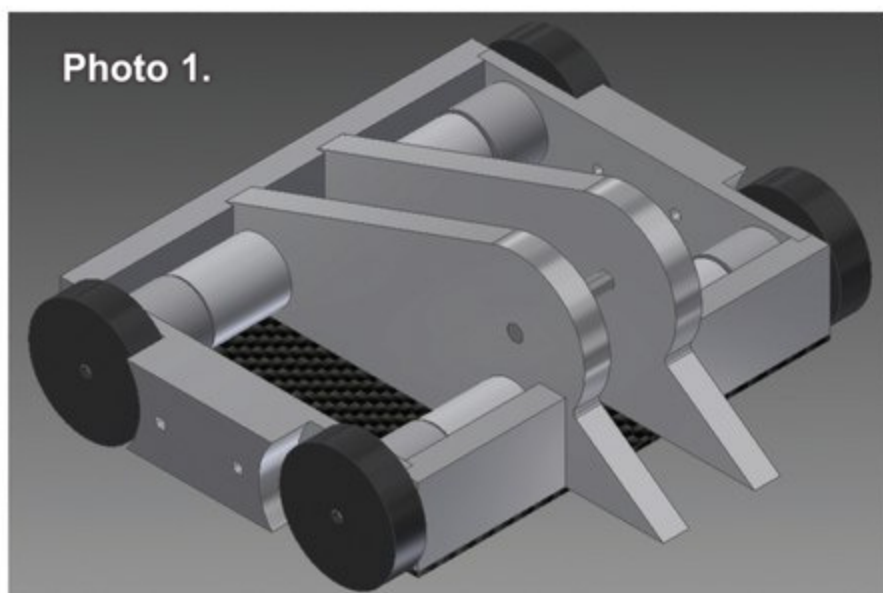


Photo 1.

LINEAR SERVOS



L12-R Linear Servo

- Direct replacement for regular rotary servos
- Standard 3 wire connectors
- Compatible with most R/C receivers
- 1-2ms PWM control signal, 6v power
- 1", 2" and 4" strokes
- 3-10 lbs. force range
- 1/4" to 1" per second speed ranges
- Compatible with VEX



L16 Linear Actuators

- 2", 4" and 6" strokes
- 10 - 40 lbs. force range
- 1/2" to 1" per second speed ranges
- Options include Limit Switches and Position Feedback

New!

PQ12 Linear Actuator

- Miniature Linear Motion Devices
- 6 or 12 volts, 3/4" stroke
- Up to 5 lbs. force
- Integrated position feedback or limit switches at end of stroke
- External position control available



Linear Actuator Controller (LAC)

- Will drive any Linear Actuator with position feedback
- Up to 24v and 4 Amps
- USB connectivity to drive the actuator with your computer
- Adjustable speed, limits and sensitivity



L12-NXT Linear Servo

- Designed for LEGO Mindstorms NXT®
- Plugs directly into your NXT Brick
- NXT-G Block available for download
- Can be used with Technic and PF
- Max. speed: 1/2" per sec.
- Pushes up to 5 lbs.
- 2" and 4" strokes



Available Now @
www.firgelli.com

aluminum backplate (**Photo 4**).

Ash landed another hit directly after that which flipped its foe over onto its back where it could still drive, but was massively impaired without a wedge. Because of the lower ground clearance of Snappy while on its back, it got one wheel caught on the lip between the arena floor tiles, giving me a perfect opportunity to get a nice hit on the backplate again (**Photo 5**).

Unfortunately, though, this was the end of the fight as the receiver pack was run down again, leaving Snappy the victor albeit with a few nibbles taken out of him (**Photo 6**).

Post event, Ash was completely unscathed, but I did learn a few things that need to be done to get it working to its full potential. First, it needs larger wheels as the ones it currently has were low enough that it got caught on the lip between the floor tiles a few times.

Secondly, it needs a new wedge as the four millimeter steel one put it barely within the weight limit, and finally a new weapon is needed. The current bar works okay, but a thicker single tooth weapon would make the hits much



Photo 4.



Photo 5.

harder, as well as having the control of a four wheel drive pusher which is what I really want from it as a competitor.

Overall, I think once Ash has these modifications it could go far in its class, and hopefully in the future I can build another bot. For now, though, I think I'll just concentrate on getting Ash working fully and getting Terminus active for demos and fun fights.

Ash is currently sitting on my desk, awaiting the modifications once the funds are available or for the next event to come around — whichever comes first (**Photo 7**). **SV**



Photo 7.

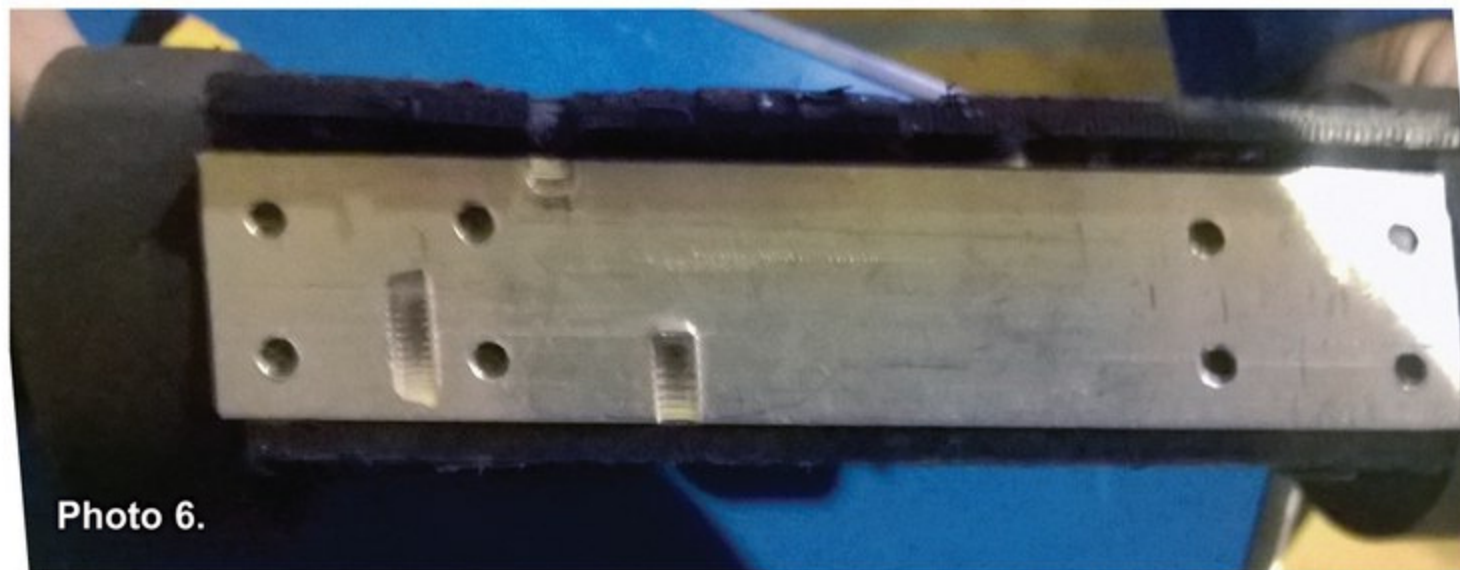


Photo 6.

EVENT REPORT:

Freeside Robot Street Fight 2015

● by Mike Jeffries

A few years ago, the original Insect arena used for DragonCon Robot Micro Battles was retired and a new arena was built. The original arena spent the next several years collecting dust in a garage. In October 2014, the arena found a new home with me. The plan was to refurbish the arena and put it back in use.

Nothing spurs on a project better than a deadline, so after some discussions with Freeside Atlanta, a date was settled on: May 9th, 2015. This would be the first event held in the arena since Labor Day weekend in 2011.

In the months between the arena moving and the event, the floor was sanded and painted, new pits were made, and the hazard was at least partially repaired. The final few details were dealt with in the traditional manner: in a rush and moments before the event started.

In total, 14 robots showed up for the event. There were six Antweights, two Beetleweights, three Hobbyweights, and three Featherweights (1 lb, 3 lb, 12 lb, and 30 lb, respectively).

In the Antweight class, the competitors were:

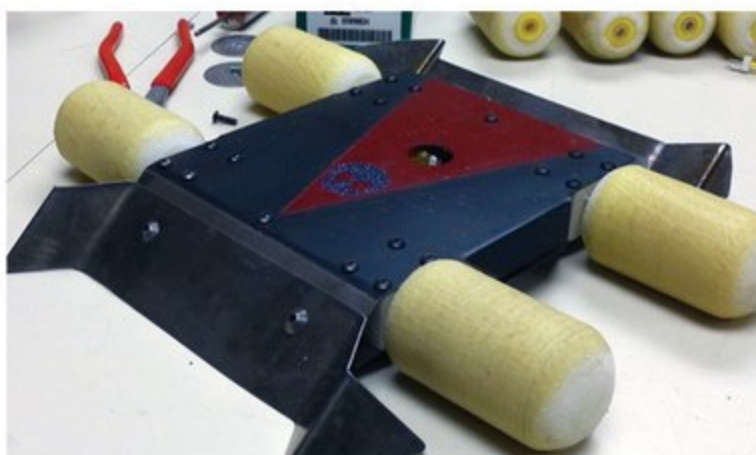
- Algos, a titanium clad vertical spinner.
- Death by Twinkies, a durable and invertible ramming wedge with Twinkie-like wheels and a combination of spring steel and titanium armor.



The arena moments before combat started.



Algos.



Death by Twinkies.

- The Wolf, an invertible wedge based on the Titan Tech Industries kit.
- Cyrano Chubbs, a two wheel drive robot with an unusual rotary lifting wedge.
- Eleos, a lightly modified FingerTech Robotics Viper Lifter

kit (reviewed in the May 2015 issue of *SERVO*).

- Gelos, a wedge with a 3D printed chassis and a hardened 3/32" thick 4130 steel wedge.

In the Beetleweight class, the competitors were:

- Torgo, a wedge shaped robot with an actuated lifting arm.
- Naked Singularity, a four wheel drive wedge.

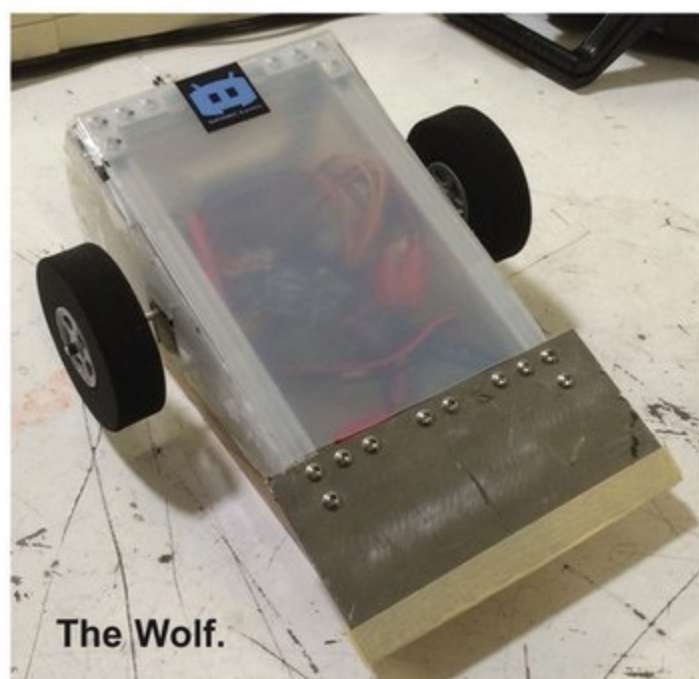
In the Hobbyweight class, the competitors were:

- Omega Force, a kinetic energy powered flipper.
- Abrasive Personality, a two wheel drive wedge with an unusual belt sander based bot capturing device.
- Hypnus, a dustpan-ish wedge with top mounted catch hooks.

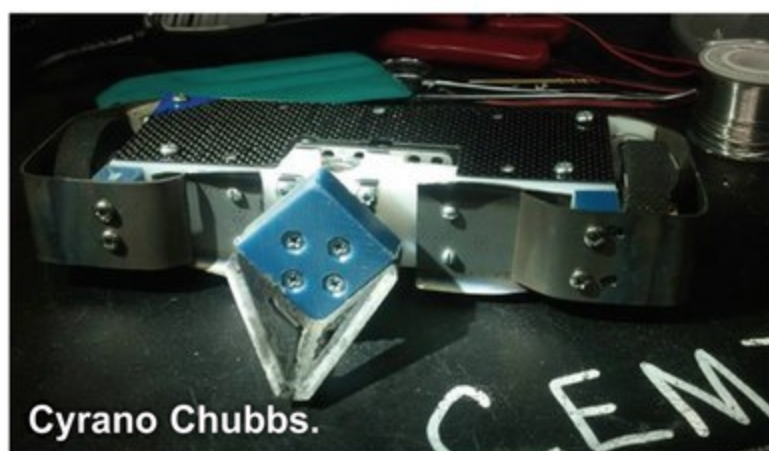
In the Featherweight class, the competitors were:

- Overthrunder, a kinetic energy powered flipper with an automatic firing mechanism.
- Nyx, a four wheel drive bot with interchangeable weapons. For this event, the lifter was used in all matches.
- Spanky, an overhead thwackbot with a custom electric chainsaw.

For the Antweight, Hobbyweight, and Featherweight portion of the events, a round robin format was



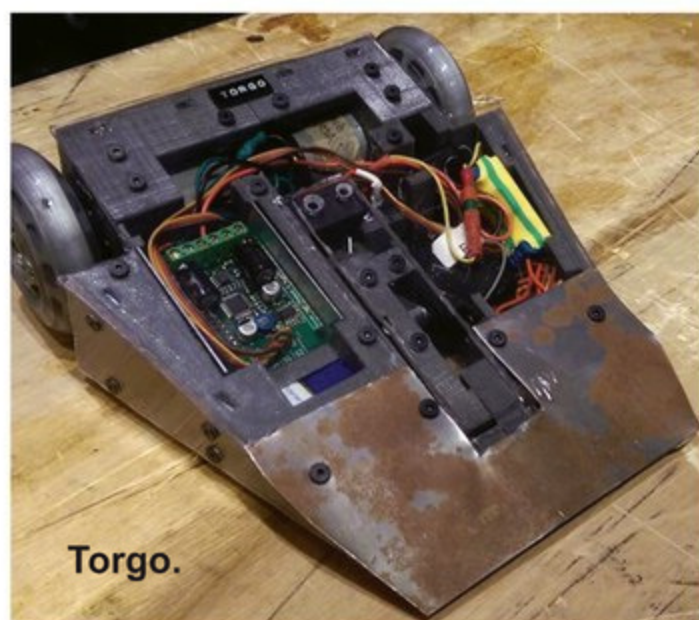
The Wolf.



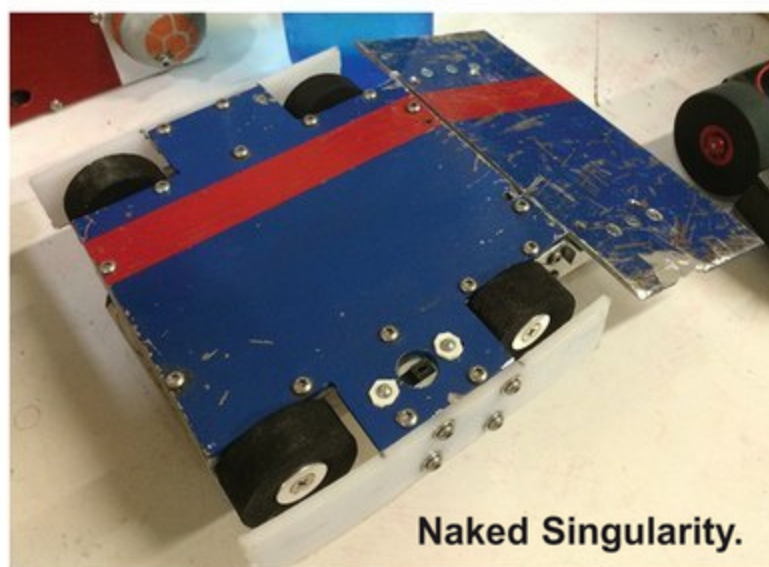
Cyrano Chubbs.



Eleos.

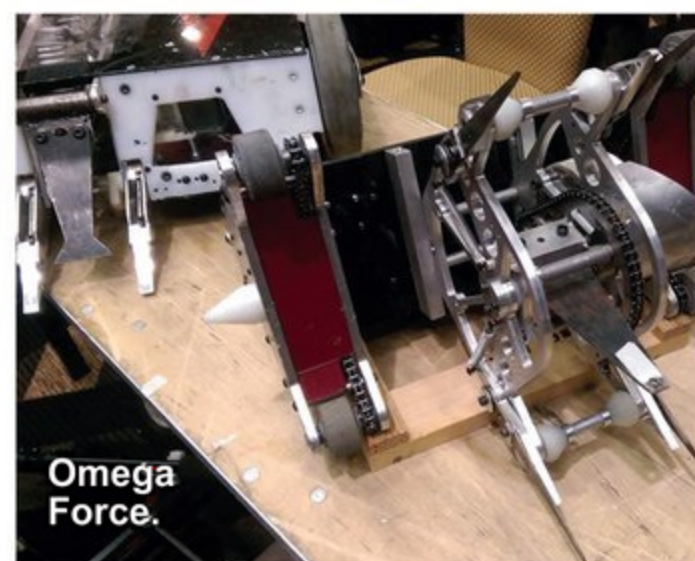


Torgo.



Naked Singularity.

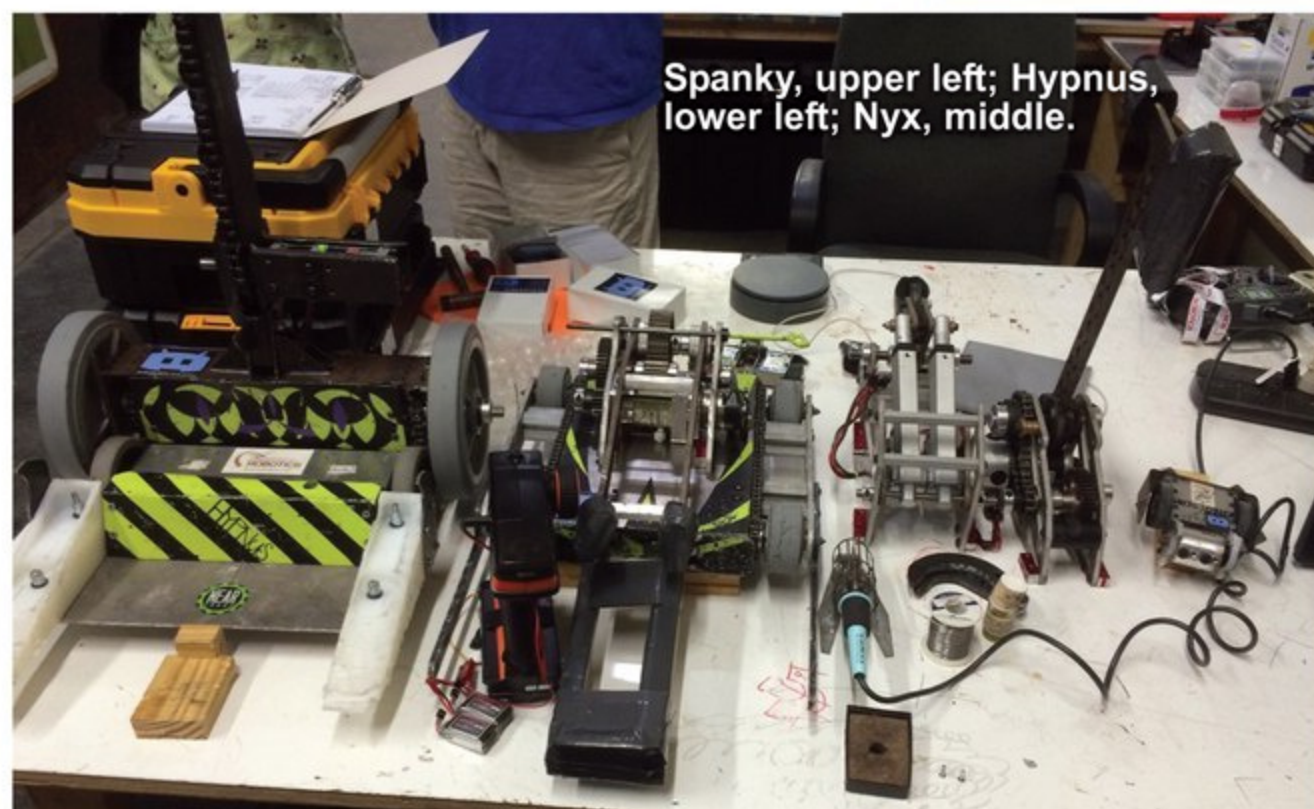
Dale Heatherington, Lewis Baumstark, and Brandon Davis contributed photos for this article.



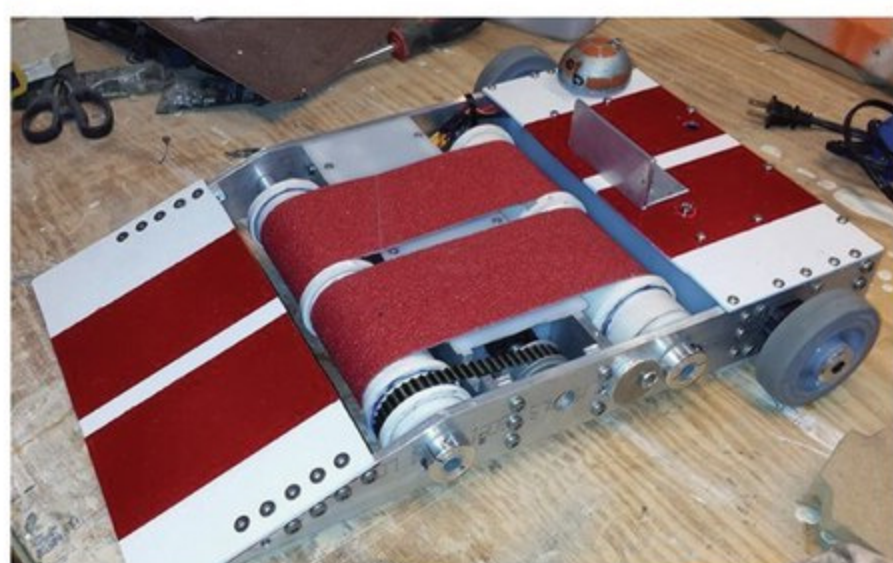
Omega Force.



Overthruster.



Spanky, upper left; Hypnus, lower left; Nyx, middle.



Abrasive Personality.

used. In the Beetleweight event, the title was decided by having the two

competitors fight three matches.

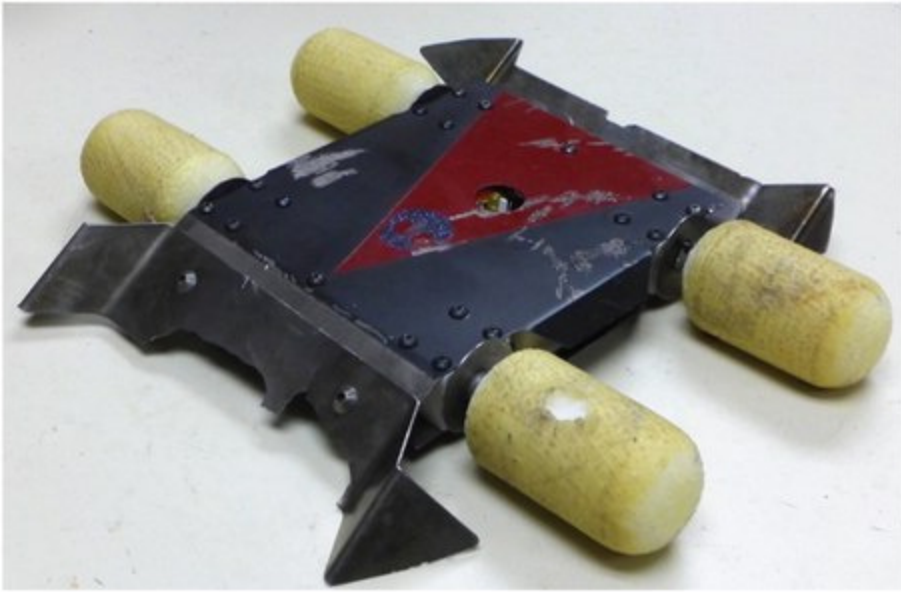
The first Antweight match was between Death by Twinkies and The Wolf. After some initial sparring, Death by Twinkies was able to push The Wolf into the pit. After that, Eleos faced Gelos.

Early in the match, Eleos was able to use its flipping arm to invert and grapple Gelos. Late in the match, Gelos lost half of its

drive and Eleos seemed sure to win until a small driving error put Eleos into the pit.

Next up was Eleos and The Wolf. Again, Eleos started strong getting in a few lifts before The Wolf was able to counter with a scoop of its own. A late driving error resulted in The Wolf straddling the pit before Eleos pushed it out.

Following that, Eleos faced Algos. Early in the match, Algos destroyed the front armor panel on Eleos and



The damage done to Death by Twinkies by Algos.



Nyx slams Overthrust into a chunk of steel bolted to the ground, stopping both robots dead in their tracks.



Nyx lifts Overthrust in the middle of the combat area.

broke the flipper arm. One of the wheels came off of Algos and put Eleos back into the fight, until it became completely high centered on its own broken weapon.

Algos then faced The Wolf and ripped the front wedge off prior to The Wolf getting knocked into the pit. After some repairs, The Wolf then faced Cyrano Chubbs in a back and forth fight that eventually saw Cyrano Chubbs drive into the pit.

Next up was Death by Twinkies against Cyrano Chubbs. Death by Twinkies quickly got under Cyrano Chubbs and put it into the pit. Cyrano Chubbs then faced Algos and had its durability thoroughly tested.

Over the course of the match, Cyrano Chubbs was hit dozens of times by the spinning weapon on Algos, and the only real damage was to the linkage attaching the rotary lifting wedge to the servo that powered it.

Eventually, Cyrano Chubbs was pitted and the match was over. After that, a broken Eleos faced off against Death by Twinkies. Eleos couldn't overcome the damage it had sustained earlier in the event and Death by Twinkies easily defeated it.

In the second to last match, Eleos faced Cyrano Chubbs. Both bots were damaged from prior fights, but Eleos was better able to manage its

damage and won the match. The final Antweight match saw the two as yet undefeated bots (Algos and Death by Twinkies) face off.

Death by Twinkies got off to a strong early start but wasn't able to sustain control. One of the early weapon impacts broke a piece off of the hardened 1095 steel wedge on Death by Twinkies and from then on, Algos was able to control the fight. Eventually, Death by Twinkies wound up in the pit making Algos the winner.

The Beetleweight matchup saw Torgo face Naked Singularity three times. All three matches were fast paced/close matches between two

Melty Brains

Bad Robot Puns

by Kevin Berry

Battery Samples
Amazing New Technology!

Free is good!

Dude, I'm not sure about this.

Joe's Bot scratches!
Unable to exhibit controlled movement in the box!

It's just sitting there twitching!

Well, Joe, you know those batteries were free of charge!



Omega Force launches Abrasive Personality into the air with its kinetic energy flipper.

powerful robots. The first match saw Torgo win when a driving error put half of Naked Singularity over the pit. The second match was much the same — a high speed drivers match that saw Torgo wind up in the pit.

With each bot winning once, a third fight had to occur. In this match,

we again had aggressive back and forth combat with some very effective use of the lifting arm from Torgo. The match ended when Torgo was able to get under Naked Singularity and put it into the pit.

The large classes faced off in a

somewhat different environment. Instead of fighting in an arena, the large robots fought in a chalked out area that consisted of some flat areas, some broken concrete, dirt, and random chunks of metal sticking out from the ground.

The first match in the Hobbyweight class saw Hypnus face off with Abrasive Personality. The bots fought as much with the terrain as they did with themselves, and after several minutes of back and forth combat, Abrasive Personality came out on top.

After that, Omega Force took on Hypnus. While the terrain still complicated things, the kinetic energy flipper on Omega Force allowed it to quickly defeat Hypnus.

In the final Hobbyweight fight, Omega Force faced Abrasive Personality and much like with Hypnus, it was able to flip the opposing robot reliably and was able to defeat its opponent.

The first Featherweight match was between Nyx and Spanky. The lifter on Nyx was able to reliably get under and throw Spanky. Between the guest driver and wild controls on Spanky, Nyx was able to easily win.

Following that, Nyx faced Overthrust in a long and brutal back and forth fight. Each robot was able to get their opponent out of the combat area and at one point both bots went out together.

After some improvised off roading and a lot of powerful slams, Nyx was able to come out on top. At the end of the fight, Overthrust experienced some electrical issues that resulted in it forfeiting to Spanky.

While the first Freeside Robot Street Fight was a fairly small event, the concept worked well and plans are already in the works for the next one. **SV**

RESULTS.

CLASS	1st	2nd	3rd
Antweights:	Algos	Death by Twinkies	Eleos
Beetleweights:	Torgo	Naked Singularity	n/a
Hobbyweights:	Omega Force	Abrasive Personality	Hypnus
Featherweights:	Nyx	Spanky	Overthrust



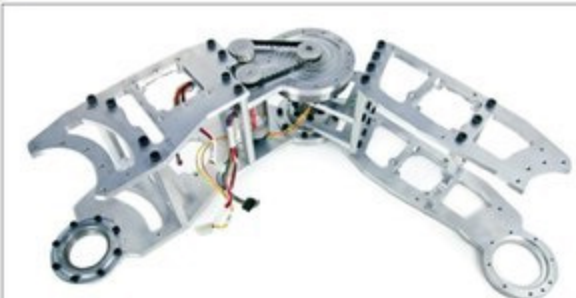
TORMACH Personal CNC Mills

Shown here with optional stand and accessories.

PCNC 1100 Series 3 starting at: **\$8480** (plus shipping)

www.tormach.com/servo

Shown below is an articulated humanoid robot leg, built by researchers at the Drexel Autonomous System Lab (DASL) with a Tormach PCNC 1100 milling machine. DASL researcher Roy Gross estimates that somewhere between 300 and 400 components for "HUBO+" has been machined on their PCNC 1100.



Little Arduino Robot Hand

(for under \$300)

By Peter Ohlmus

In late 2012 while on holiday, I made an inexpensive little robotic hand (**Figure 1**) using bits and pieces I found at various shops, along with the newly available and very cool Arduino microcontroller.

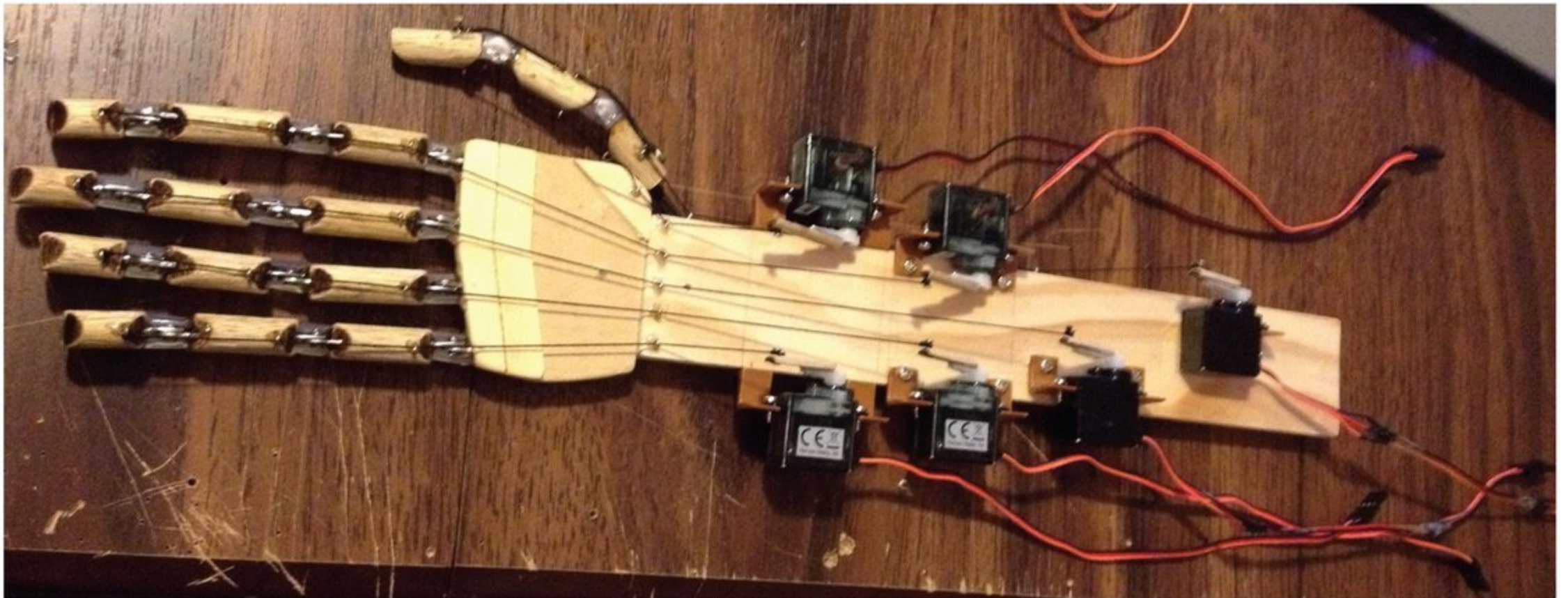


Figure 1. The little Arduino robot hand.

I have always loved playing with electronics and pulling things apart to see how they work. I have been assembling and playing with radio controlled vehicles and servos since I was about eight. So, I was quite excited when I noticed an “Arduino Experimenter’s Kit” at my local electronics store. After some online research, I was surprised and motivated by the possibilities I could pursue.

Initially, I thought about making a bipedal robot using micro servos as I’d been watching those classic cool Taiwanese robot fighting videos on YouTube. This changed quickly, after I began to imagine the prospect of mimicking hand movements, and achieving this cheaply using common materials and fast reliable servos. Thus, a da Vinci inspired robotic hand project was conceived!

It was my first play with the Arduino and writing code to control servos. To an experienced person, the wooden hand might look simple, but it demonstrates some

fundamentals of servo control, Arduino programming, and mechatronics. Plus, it’s a fun adventure sourcing your own materials, thinking about solutions, putting it altogether, and seeing the final working product — as well as thinking of improvements and changes along the way.

All the parts used can be easily found in hardware and haberdashery stores, as well as local supermarkets. Once you’re comfortable with programming the Arduino, assembling and configuring it all is a satisfying experience.

How I Began

I didn’t spend a lot of time planning the project, and my first sketches and designs were quite basic; sketches of the finger joints and the size of the hand, along with placement of servos and the string I would use as tendons.

As the idea developed, I found it much easier just to source and make my own parts, and adjust them to work



Figure 2. Tea infusers provided the hinges (knuckles).

as I progressed. I basically modeled the wooden hand loosely on my own hand, with each finger segment roughly the same size as my own. If you're looking to make something similar, I'd suggest you measure each segment of each finger of your hand to get a scale model.

I'm glad I didn't have a 3D printer, as drawing, cutting, drilling, gluing, throwing out, and starting again adds to the enjoyment and creativity! Some good music or a good documentary to listen to in the background, all the requisite tools, enthusiasm, and plenty of time, and you're good to go!

To start with, I chose a one meter length of 10 mm diameter dowel to act as the finger segments, cutting them to suit my sketches using my faithful Dremel Multipro. The blackened ends of the finger joints you see in my YouTube video were from the Dremel cuts — basically burning the wood as it cut.

Now, I needed knuckles, so I began looking in hardware stores for cheap but effective hinges which could join the finger segments together and enable them to move. I looked at small hinges, curtain rod connectors, scissors, and other potential donors, but it wasn't until I was in the local supermarket and I saw a tea-infuser hanging on the wall that I knew I'd hit paydirt (or hingedirt)!

They were \$2 each, and each infuser had two pressed metal 'hinges' — two pieces of metal with a hole and pin that had been shot peened at one end. So, I bought 15 of them for \$30 (to the eye-raised surprise of the checkout operator!), and low and behold oodles of infusers awaited their fate (**Figure 2**).

I checked all of the hinges, and out of all 30 only about 18 moved smoothly and didn't jam. I mean these are



Figure 4. Elastic strip to provide resistance when fingers bend.



Figure 3. Eyelets to guide tendons. Note the 'cut-and-ready' pair at right.

average quality pressed steel and not designed with tight tolerances in mind or to flop around smoothly.

The 18 that passed the test were more than enough for the three knuckles I needed for each finger.

A visit to the local haberdashery store and I found suitable thick cotton thread for tendons and some very small metal eyelets to be transformed into tendon guides (**Figure 3**).

I also found some narrow black elastic (like you would find in a pair of pants) which I wanted to use on top of each finger joint to provide resistance — to return the finger to an 'open' position (**Figure 4**).

This elastic would also be applied to the thumb assembly during rotation/swiveling.

The 'opening' and 'closing' finger movement is actually termed 'flexion' and 'extension.' Bending the fingers toward the palm is flexion, or what I'm calling closing the finger; bending the fingers out straight away from the palm is extension, or what I'm calling opening the finger.

In between the finger segments (top of the hinge/knuckle), I used pieces of cut silicon tube. This provided an improvised form of cartilage to stop the finger segments bending back too far, and to also provide a small



Figure 5.
A nearly complete finger; note the 45 degree angle cuts on the lower sections so the finger can fully close.

amount of natural cushioning.

For the palm and forearm sections, I used pine wood, MDF, and a few sliced up paddle pop sticks.

As far as moving the fingers went, I figured I'd need six servos: one for each finger and two for the thumb.

To control it, all I wanted was four inputs to give the hand four different functions, so I opted for a basic control board PCB (printed circuit board) that would have four buttons (for the inputs), and would be powered by a separate 5V power supply for the six servos.

Finally, there was the shiny new Arduino Uno which would be powered by my laptop.

This collection of bits and ideas all made a lot of sense and I jumped right into it — exactly what the enthusiastic tinkerer does on holidays!

I was now ready to build and program the little wooden hand.

Putting It All Together — Assembly and Soldering

I had the sketches of the finger segments done, so it was just a matter of marking the dowel and cutting all the segments. I cut the lower end of each palm-facing segment at 45 degrees, so the finger could close fully without the ends of each segment touching and not being able to travel further (**Figure 5**).

I needed to cater to the knuckle/hinge, so there would be a gap of about 10 mm between two facing segments,



Figure 6. Hinges cut out from tea infusers and ready for use.

after which I would drill holes in the finger segments.

The next step was to cut the hinges out of the tea infuser, leaving approximately 5 mm of shaft on either side of the hinge (**Figure 6**).

Use a drill bit that has a slightly greater width than the exposed tea infuser shafts, and then drill a hole into each end of each segment (except the finger tips). This is where the infuser shafts will go, allowing a small amount of space for the epoxy. Apply some epoxy to the holes and insert the shaft into the dowel. Make sure the hinges all line up as you don't want to have wonky fingers bending at unusual angles. Also, be conscious of this placement especially if using quickset epoxy.

Now, it was time for the cartilage pieces. These need to be cut the same size as the gap between the finger segments (approximately 10 mm). Cut a piece of silicone the same length as the gap, then cut it in half.

You will want to epoxy one end of the silicone to the top half of one of the finger segments only; you don't want to glue the two segments together (**Figure 7**).

I then stretched the black elastic band pieces across the top (non-palm facing) of each finger segment, and held each end in place with rubber bands (you can also use

Advice/Tips

- Get a fresh supply of epoxy and super glue.
- Use strong string/thread for the tendons.
- Make sure the servos are mounted tight to the forearm.
- Let the epoxy cure.
- Test the code and servo on their own BEFORE connecting them to a finger.
- Enjoy yourself and learn about mechatronics.
- Let the creative juices flow.



Figure 7. Tendon guides and cartilage cushion/stop added. Tip of finger is at left.

small alligator clips). Then, I added a tiny drop of super glue under the elastic where it met the wood to hold it in place (**Figure 8**).

When set, I tested there was the right amount of flex by opening and closing the finger, and if it felt right, I applied epoxy resin to the underneath of each end of the black band, trimming the excess off each end after the epoxy had set. If it didn't feel right, the super glue bond was easily broken, allowing me to stretch it more or less and apply super glue again. I used Blu Tack to stick the now hinged-up finger to the table, while I placed and glued the elastic.

Now, on to the palm and forearm. I cut the piece of pine to 300 mm in length which was plenty for the forearm, then I traced my original palm sketch onto the piece of MDF and cut it out. Paddle pop sticks were added to the palm to simulate padding.

After lining up the end of the fingers with the end of the palm, I marked and drilled holes for the 'finger-to-palm' hinge, which was the last exposed hinge (**Figure 9**).



Figure 8. All four fingers with elastic resistance strip applied and curing.

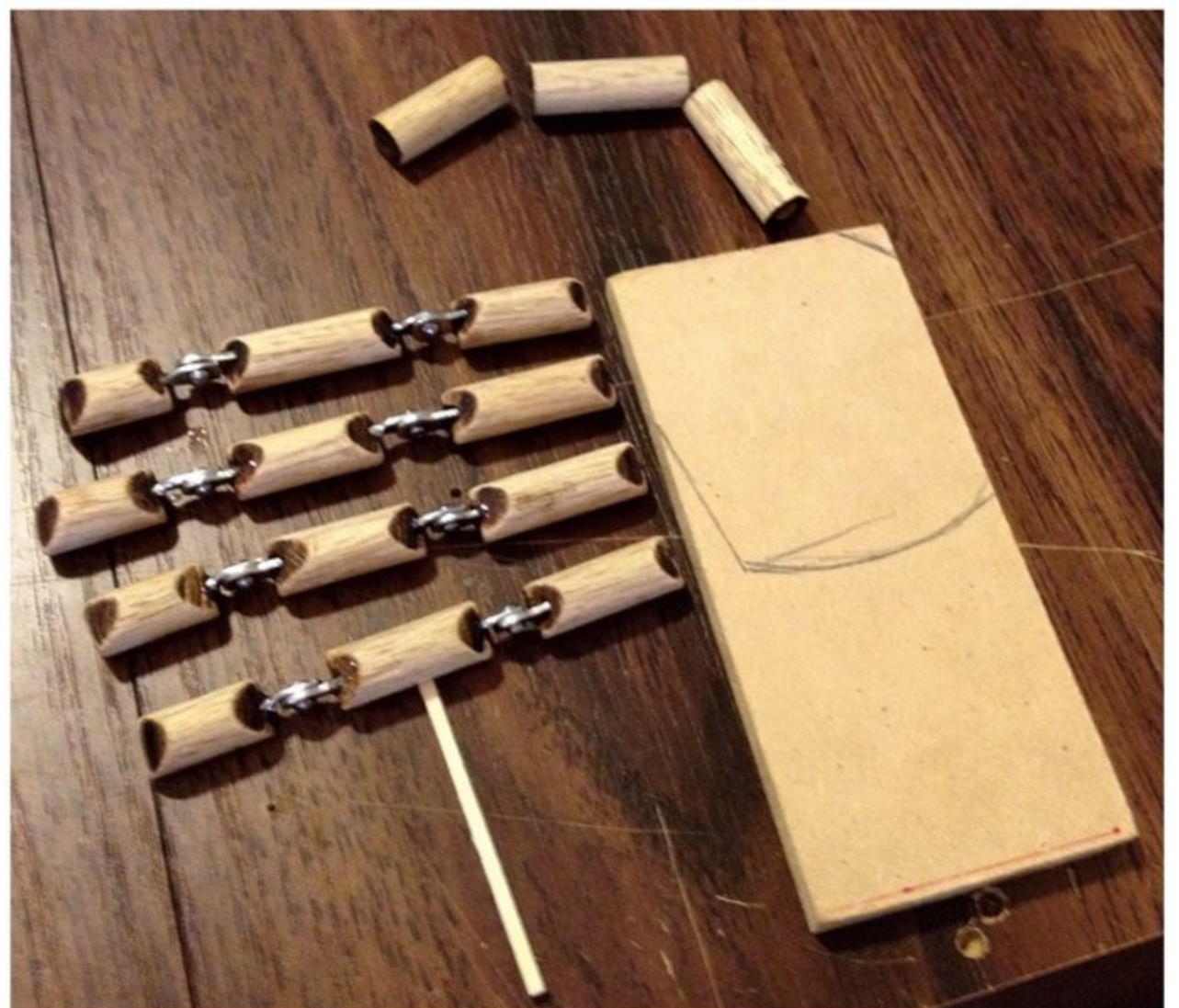


Figure 9. Earlier stages. Tracing out the shape of the palm on a piece of MDF.

Again, I used Blu Tack to hold the fingers in place while applying epoxy to the palm holes and hinge ends.

After completing the palm and the forearm, I attached them together with a flat piece of steel and moved onto the thumb connection point on the palm.

I had to figure out how my real thumb actually moved,



Figure 10. The mechanics are complete. The final step was applying the thumb assembly return elastic.



Figure 11. Tendon guides on the forearm for all five digits. Not shown are the eyelets to swing the thumb assembly up.

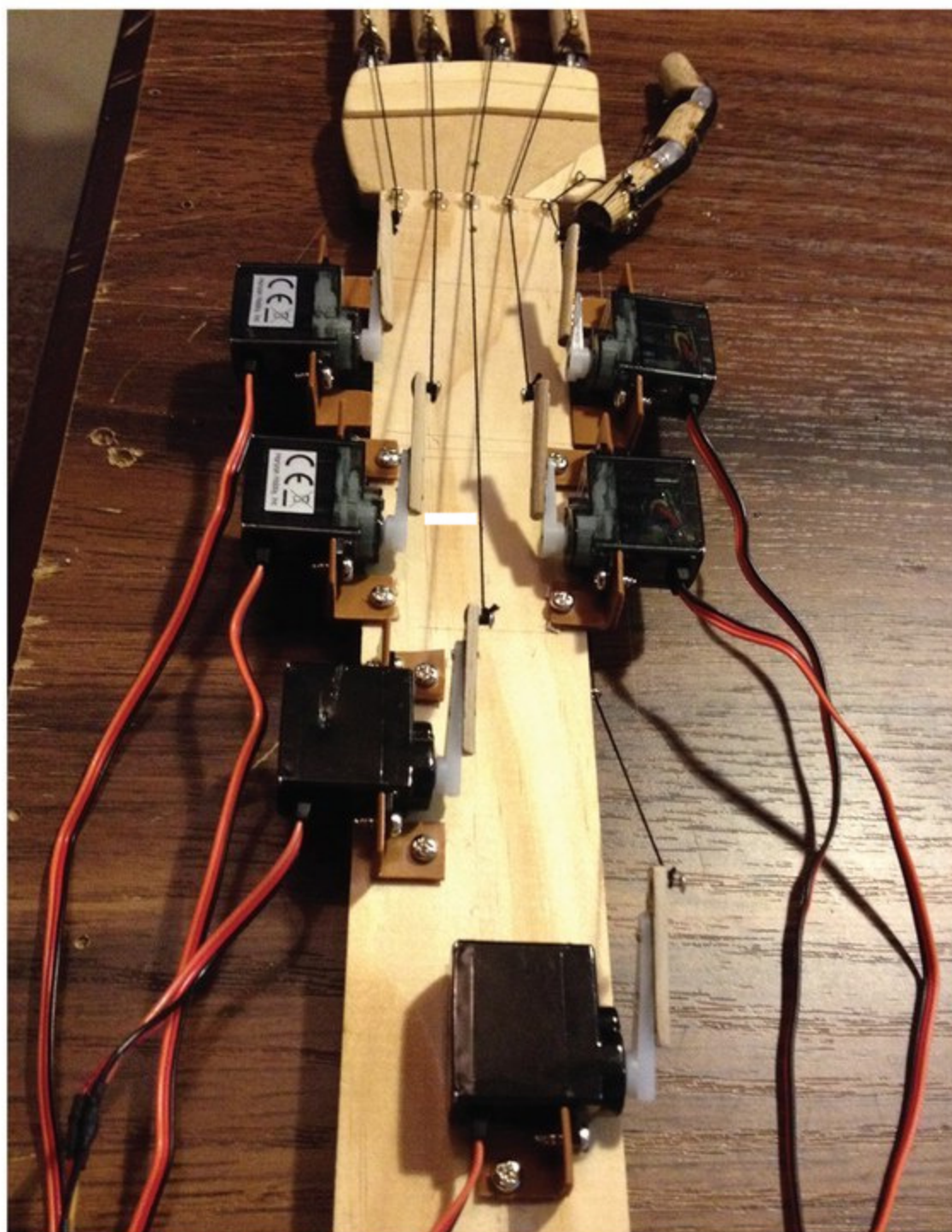


Figure 12. The complete mechatronic assembly. The servo at the bottom of the picture controls the thumb assembly swivel, with the tendon running underneath the servos on the right.

and ended up attaching the completed thumb assembly at a 45 degree angle to the bottom corner of the palm. I drilled a hole in this point of the palm and also through the



Figure 13. Tendons all threaded and ready. The anchor point to swivel the thumb assembly can be seen at the bottom of the thumb.

last thumb segment, then screwed them together with a small amount of play so the thumb assembly could swivel smoothly.

Elastic for the thumb was two-fold; where the thumb assembly itself has strips of elastic for each segment of the thumb, a single separate piece of elastic is also required to return the entire thumb assembly back to the open palm position.

I attached this piece of elastic to the thumb segment closest to the palm, then anchored it to the forearm. Two small pieces of metal were then screwed to the back of the palm to stop the thumb assembly from continuing to rotate (Figure 10).

Next in line were the tendon guide eyelets. For these, I basically just drilled small holes at the lower end of each finger segment, then cut an eyelet in half and glued the



Figure 14. Testing the control board PCB connected to an Arduino Uno, with all six servos.

caught on the other servos.

One of the most important steps was mounting and aligning the servos. Each servo horn had to line up with the final tendon guides. So, I placed each servo and eyeballed the position, then made basic mounts using right angle plastic brackets, and drilled and screwed them into place.

You'll note the staggered arrangement in the picture — this is because they couldn't all be lined up next to one another or they just wouldn't fit (**Figure 12**).

Once the servos were locked down, I tied the string to the fingertip eyelet of each digit, then fed the string down through all the eyelets until it reached its respective servo horn (**Figure 13**). With all hand parts now assembled, I moved to soldering all the components to the control board PCB.

The control board PCB is quite easy, as all you need to do is solder six sets of three header pins (break-away header pins) to plug the servos into, four momentary pushbuttons, four 10K ohm pull-up resistors for the buttons, and wiring to go to the Arduino (**Figure 14**).

Finally, pick a DC jack that fits a 5V power supply. In my case, I had an old Canon camera USB battery charger sitting in a box. I won't go into the detail of soldering the control board, but the supplied Fritzing diagram will guide you on the layout (**Figure 15**).

With assembly and soldering complete, I sat back and thought, cool! Now, let's get this moving!

Programming the Arduino and Testing

Once I built the mechanics and soldered the control board PCB, it was time to do the coding. I used four pushbuttons on the control board so I could activate the functions I wanted. Each button required a 10K ohm pull-up resistor between the button legs (both positive and ground connections) and jumper wires going from each button to four available digital pins on the Arduino.

I had thought about using flex sensors in a separate glove arrangement instead of buttons to control the functions, but that would have required more code and putting together a suitable glove. I still liked this idea, though. So, the four pushbuttons would trigger four basic functions, each containing a set of actions:

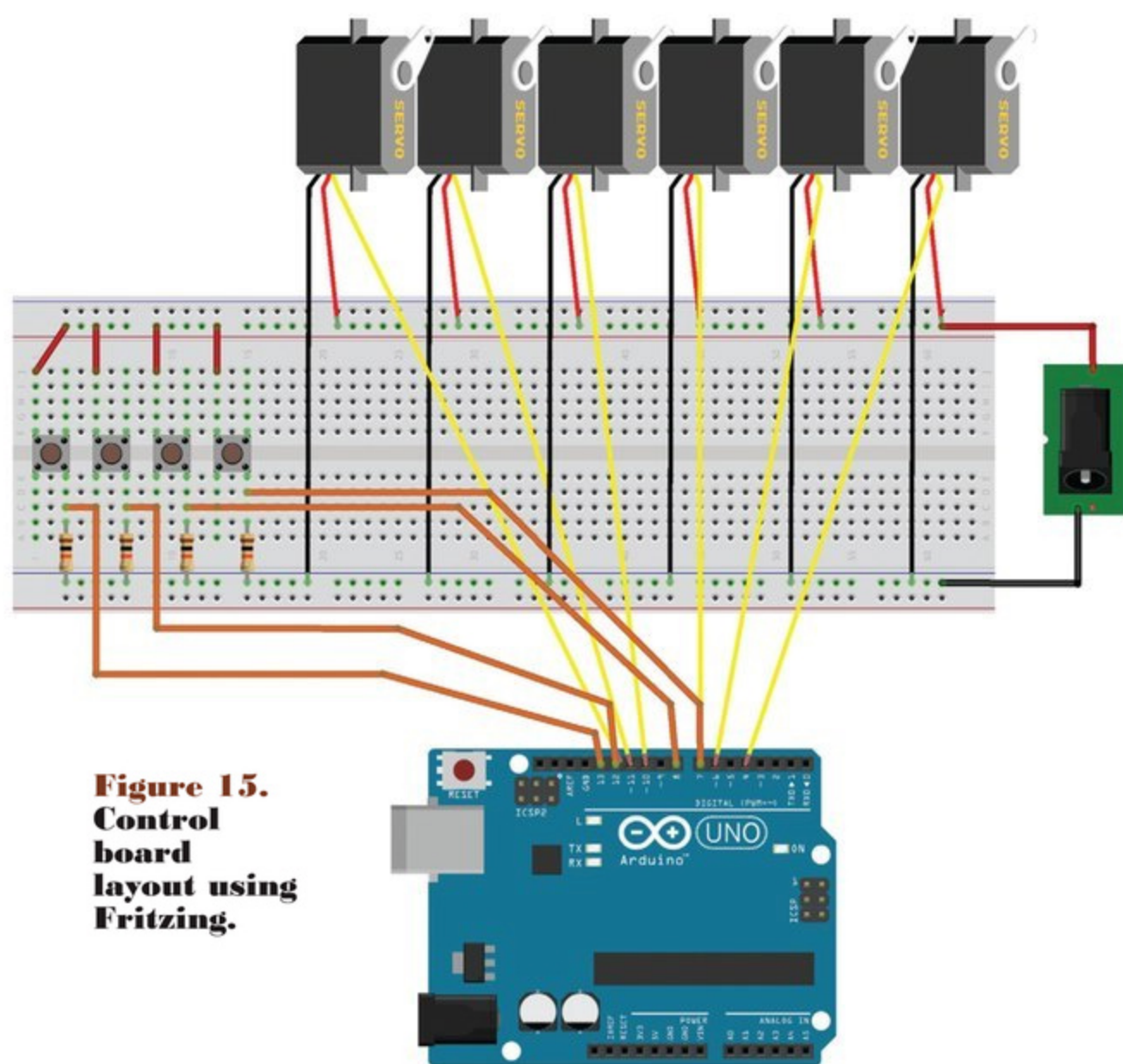


Figure 15. Control board layout using Fritzing.

single eyelet loop into the hole. I repeated this on the palm, but had to align these final guides with the servo placement on the forearm (**Figure 11**).

An eyelet was added at the thumb segment closest to the palm. This would provide the anchor point to swivel the thumb assembly up and down. Another eyelet was added to the end of the forearm where it met the palm, right beneath the point where the thumb assembly was attached. This second eyelet would guide the thumb swivel tendon directly back to the servo.

A third eyelet was added further down the side of the forearm to keep the thumb swivel tendon from getting

- **Button A** = Open and close all fingers and thumb together
 - Set slow speed
 - Close the four fingers at the same time
 - Swivel thumb up and close it
 - Open thumb and swivel down
 - Open the four fingers at the same time
- **Button B** = Move each finger then thumb independently, slow then fast
 - Set slow speed
 - Close and open finger one
 - Close and open finger two
 - Close and open finger three
 - Close and open finger four
 - Swivel thumb up, close and open, swivel down
 - Set fast speed
 - Close and open finger one
 - Close and open finger two
 - Close and open finger three
 - Close and open finger four
 - Swivel thumb up, close and open, swivel down
- **Button C** = Finger and thumb pinch – “precision” grip of something
 - Set slow speed
 - Swivel thumb up
 - Close finger one and thumb at the same time
 - Open finger one and thumb at the same time
 - Swivel thumb down
- **Button D** = Close and open fingers and thumb alternately and fast
 - Set fast speed
 - Close and open finger one and finger four at the same time
 - Close and open finger two and finger three at the same time
 - Close and open finger one and finger three at the same time
 - Close and open finger two and finger four at the same time

After attaching the tendon string to the first finger and threading it through all the eyelets, I would pull the other end of the string – which closed the finger – across a piece of paper and would mark the position of the end of the string. Then, after releasing the string with the finger returning to open, again I marked the end of the string on the paper.

This allowed me to measure the distance the string travelled, and therefore how far the servo needed to travel to open and close each finger.

This measurement would be used to fine-tune the servo travel angle in the code for each finger.

Using the example Arduino ‘sweep’ sketch, I connected a single servo to the Arduino, secured it to the table alongside the above measurements, and ran the program.



Figure 16. Servo horn extensions — used to increase the servo travel angle horizontal distance — so fingers could be fully closed and opened.

Watching the tip of the servo horn as it moved back and forth, I could clearly see that the travel angle I needed was a full 180 degree sweep. However, the length of the servo horn was too short as it didn’t travel the full horizontal distance needed. So, I made a small wood extension for each servo horn and made sure the distance they travelled once attached to the servo matched the horizontal measurements (**Figure 16**)

With those key measurements done, it was time to write some code. I started with the basics of attaching the buttons to digital pins:

- `buttonPinA` = digital pin 13
- `buttonPinB` = digital pin 12
- `buttonPinC` = digital pin 7
- `buttonPinD` = digital pin 2

The servos were then connected to the Arduino as follows:

- `myservo1.attach(11);` Pinky finger attached to PWM pin 11
- `myservo2.attach(10);` Ring finger attached to PWM pin 10

Parts List

- Piece of 10 mm x 50 mm x 400 mm pine (forearm)
- Piece of 10 mm x 60 mm x 70 mm MDF (palm)
- 1 m long piece of 10 mm diameter wooden dowel
- Right angle plastic bracket (to mount servos to forearm)
- Paddle pop sticks (not required)
- Silicone tubing x 200 mm
- Black elastic
- Strong thread/string
- Metal eyelets
- Tea infusers
- Arduino Uno (or similar)
- Jumper wires
- Pushbuttons
- Resistors (10K ohm)
- DC jack
- Breakaway header pins
- Servos x 6 (I used 7.6 gram DS76 digital sub-micro servos)

Tool List

- Drill and drill bits
- Alligator clips
- Small rubber bands
- Epoxy
- Super Glue™
- Blu Tack
- Pliers
- Dremel or hack saw
- Soldering iron and solder
- Arduino IDE (v1.0.4)

- `myservo3.attach(9);` Middle finger attached to PWM pin 9
- `myservo4.attach(6);` Pointer finger attached to PWM pin 6
- `myservo5.attach(5);` Thumb (swivel up/down) attached to PWM pin 5
- `myservo6.attach(3);` Thumb (close/open) attached to PWM pin 3

Each button press would be linked to a statement in the code: if (*digital Read*) (*button Pin A*) == *HIGH*). Or, if button A was pressed, then execute a *for* loop containing a group of `myservo1.write(n)` statements to move the servos and thus the fingers.

Again, looking at the example 'sweep' sketch, the *for* loop contains the variable *pos* (position). It is this one variable that is used to assess the condition to stay in or exit the loop, as well as incrementing/decrementing the counter. More importantly, this variable represents the servo position value for the `myservo1.write(n)` statement, where *n* is replaced with *pos*. As the loop runs, the *pos* value (which begins at 0) is incremented, and thus the position of the servo is changed one degree at a time by using the `myservo1.write(pos)` statement.

So, basically the code was repeated until a condition was met, then stopped. In this case, the condition was if the value of *pos* is equal to the maximum travel angle position value — which was 180 — then stop.

All the above code represented moving the finger from the open to the close position. To move it back to open was easy — it was just a matter of adding a second loop, but starting with a value of 180 and counting in reverse (decrementing the counter). This second loop followed the first, resulting in the full close and open movement.

I watched how fast the servo rotated based on the code, and how by including the time delay function `delay()`

after each individual or group of `write()` position movements, the rotation speed could be increased or decreased, and thus the speed of the fingers closing and opening could be changed. For example, `delay(5)` would cause the program to pause five milliseconds before continuing to the next `write()` position statement.

I attached the string to the servo horn and executed the code. The results were in: It worked great as the first finger closed and opened smoothly, and as planned. Woo hoo, I gave myself a high five!

After this success, it was a matter of replicating the code and adjusting it slightly for each finger until they were all done. Be sure to test each finger independently, and when satisfied you can combine the code so multiple fingers move at once. The thumb was a little more challenging, so I chose a basic movement model:

- Swiveling the whole thumb up
- Closing the thumb
- Opening it, then
- Swiveling back down to lay flat with the palm

It took some time to get both servos working together as I wanted, but by breaking the two movements into separate actions, eventually the final movement was smooth and integrated.

The coding was complete, testing had identified some fine-tuning that was needed, and — best of all — I had a cool working robot hand with four functions. I was quite happy and knew that if I could accomplish this, then I could do so much more using the Arduino, great little micro servos, and a lot of bits and pieces!

Behold, the Little \$300 Wooden Robot Hand!

Having succeeded with the project, I wanted to show the world! I created a few basic videos of it in action, merged the results, and uploaded it to my YouTube channel. It's been really popular, with lots of comments and questions — all of which I try my hardest to answer.

In demonstrations, people are quite taken with it and their questions and delight make the whole experience even better. I enjoy explaining how I built it and how it functions, even inspiring friends at work to go out and experiment with an Arduino and the work-horses of mechatronics — the servo. Overall, I thoroughly enjoyed my foray into mechatronics. Although I haven't fired the hand up in a while and have since repurposed the servos, it still has a prime position on my workshop shelf. What's really great is all of the components and parts came to under \$300!

Challenges and Obstacles

Given the simple nature of the components, there

really weren't many major problems or challenges.

The first problem I encountered was the servos I'd harvested from an old four-channel RC helicopter. They were basic unbranded micro servos, and although they're not bad to begin with — especially considering the low cost involved — they're designed to be harnessed to the helicopter frame, so the servo housing base plate is not secured to the rest of the housing with four screws like you would usually see.

This meant the plate kept popping off, grease inside was leaking out, and they really weren't secure like regular servos. Initially, I put a rubber band around the whole housing which stopped the insides from popping out. However, despite this, in the end they just weren't up to the task — a few stripped gears and a failed motor appearing as time went by.

As I have been flying e-Flite Blade RC helicopters for several years, I knew their micro servos were reliable, strong, and relatively cheap. So, I went ahead and bought six servos to replace the others at a reasonable \$17 each.

The next hurdle was the Arduino. My coding experience with C and VB was many years ago and totally removed from microcontrollers. The ultimate end result of this project was multiple servos doing multiple things at the press of a single button (four buttons/functions in total), so it was a steep learning curve.

However, after discovering the online community, downloading a few sample sketches, and blowing away the coding cobwebs in my head, I wrote my first basic sketch to move a servo using PWM (pulse width modulation).

It was really easy, and the code is simple to understand. So, all in all, the challenges were fun, relatively easy to work through, and not daunting "can't sleep at night" problems. The beauty of the Arduino is the massive community online with help just a search away, along with

Resources

www.youtube.com/watch?v=t52edTD9RA0
www.youtube.com/user/petethedreamer/videos
Arduino.cc; Adafruit.com; Sparkfun.com
www.e-fliterc.com/Products/Default.aspx?ProdID=EFLRDS76

great write-ups on various websites such as **Arduino.cc**, **Adafruit**, and **SparkFun**. Plus, there's the articles available in issues of *SERVO* and also *Nuts & Volts* (www.nutsvolts.com).

Newer More Challenging Versions

Fast forward one year and I built an all metal version of a similar hand. This time, I used spring wall anchors/toggles for knuckles and 10 pound braided fishing line for tendons. Once again, I used parts that could be easily found at a hardware shop along with the same Arduino Uno and e-Flite micro servos.

The metallic hand looks a bit cooler — in a Terminator way — like an exoskeleton. It has full wrist rotation and wrist flexion/extension movement, along with finger splaying/adduction (although I never finished that function).

I am now working on version 3, made mostly of 3D printed parts, and using linear actuators I designed myself and am currently configuring. It is much more efficient, precise, and compliant, and the pipe dream is to make an inexpensive, capable, and modular mechatronics kit for kids and hobbyists.

If that works out, and as the design improves, my ultimate vision is to make affordable, easily customized prosthetics for lower limb amputees. **SV**



PBP3

PICBASIC PRO™ Compiler 3.0
BASIC Compiler for Microchip PIC® microcontroller

FREE
Student - ~~\$49.95~~
Silver Edition - \$119.95
Gold Edition - \$269.95

A world class BASIC Compiler for rapid development of Microchip PIC® microcontroller based projects.

- Lightning Fast
- Generates Optimized, Machine Ready Code
- Easy enough for the hobbyist, strong enough for a pro.
- This is a full-blown development tool that produces code in the same manner as a C compiler (without the pain of C).
- It is very easy to learn and understand.

Download a **FREE** Trial Now!
www.PBP3.com

PICBASIC and PICBASIC PRO are trademarks of Microchip Technology Inc. in the USA and other countries. PIC is a registered trademark of Microchip Technology Inc. in the USA and other countries.

DIY ANIMATRONICS

IT'S ALL IN YOUR HEAD!

By Steve Koci

Giving your characters a voice and some head movement

One of the most popular projects for the Do-It-Yourself prop builder is constructing a character that has the ability to talk using a selected voice track. You can then take it to the next level by adding the ability for the head to move in three axes — to rotate, tilt, and nod. These characters can then be utilized in a wide variety of situations: to greet customers in a store; to educate and give information at a museum exhibit; to entertain guests at a school carnival; to be used as props in a Halloween or Christmas display; or to demonstrate a product.

We're going to look at several different methods to accomplish this.

There's More than One Way to Skin this Cat

We will start off with detailing the steps needed to hack an already built prop in order to bend it to our needs. We will tear into an off-the-shelf prop and look at the required steps to alter it, so that we can control its movements and its audio track.

The second example will start with a plastic skull kit and add in the components to allow it to move its jaw in sync with your audio track, as well as to have the ability to move in three axes. Even if you have no interest in displaying a bare skull, I'll show you how we use them as a base to create some extraordinary characters.

The three-axis talking mechanism from Monster Guts is designed to fit inside the plastic skull which becomes an anatomically correct platform on which to sculpt.

The final project will explore a method for building a structure on which a mask can be attached. It will use the Actobotics components available from ServoCity to simplify the build. However, if you are interested in a more DIY project, check out the model designed by Halloween2Go at <http://halloween2go.com/volume1p1.htm>.

This is where I got the original project concept, and I've constructed several of these that I am currently using in my display. The Actobotics parts greatly simplify the build, eliminating the need to construct all the necessary components from scratch. You can simply order the parts

Post comments on this section and find any associated files and/or downloads at www.servomagazine.com/index.php/magazine/article/august2015_Koci.

you need and quickly put your prop together using only a screwdriver, an Allen wrench, and a few common woodworking tools.

What is this Going to Cost Me?

The cost will vary widely depending on which project you are planning to undertake. If you will be hacking a toy you already have, for example, the only cost may be for the controller. However, if you need to purchase all the components for one of the complete projects, the cost will be in the \$300 range.

Can You Hear Me?

The simplest way to get a talking character up and running is to start with something where much of the work has already been done. Finding an existing toy or prop and adapting it to fit our needs will greatly reduce the time and effort required to achieve our goal (**Figure 1**).

The primary problem with this approach always seems to be that the supplied controller wasn't easily hacked and doesn't allow us to replace the provided audio track with one of our own. In addition, instead of using servos to provide the movements, they utilized small DC motors.

A talented group of haunters from Haunt Forum attacked this issue and came up with a board that lets us control the motors as we wish. I took their design and added an onboard stereo player to supply the audio tracks and a PICAXE chip to handle the trigger duties. I now have a single board solution for hacking these props.

Once you have selected your character, you will proceed with a bit of simple surgery. The wires connecting the jaw motor will need to be removed from the circuit board and attached to the new controller.

If you're interested in following the progression of the board development process, you can check out the full thread at www.hauntforum.com/showthread.php?t=26340.

The circuit works by turning a MOSFET on and off as it responds to the tone track created in Audacity. More info on this process can be found later in this article, as well as in the Instructable on the board put together by one of the originators,



Figure 1. The poor monkey selected for surgery.

Jeff Haas at www.instructables.com/id/Modify-A-Talking-Skull-To-Say-Anything-You-Want/.

For this article, I originally planned to hack a broken monkey toy and replace the provided circuitry with something similar to the one referenced above (with my modified version), but things did not go as planned. The monkey used a single motor which was attached to a complex gear setup to provide multiple movements (**Figure 2**). This meant that I would be unable to isolate the jaw by simply hacking its separate motor. So, it was on to Plan B.

My other option was to replace the existing motor with a servo and use a circuit like the one developed by Scary Terry to drive the jaw. Again, I've tweaked his design a bit



Figure 2. Wow! There's a lot going on in there.

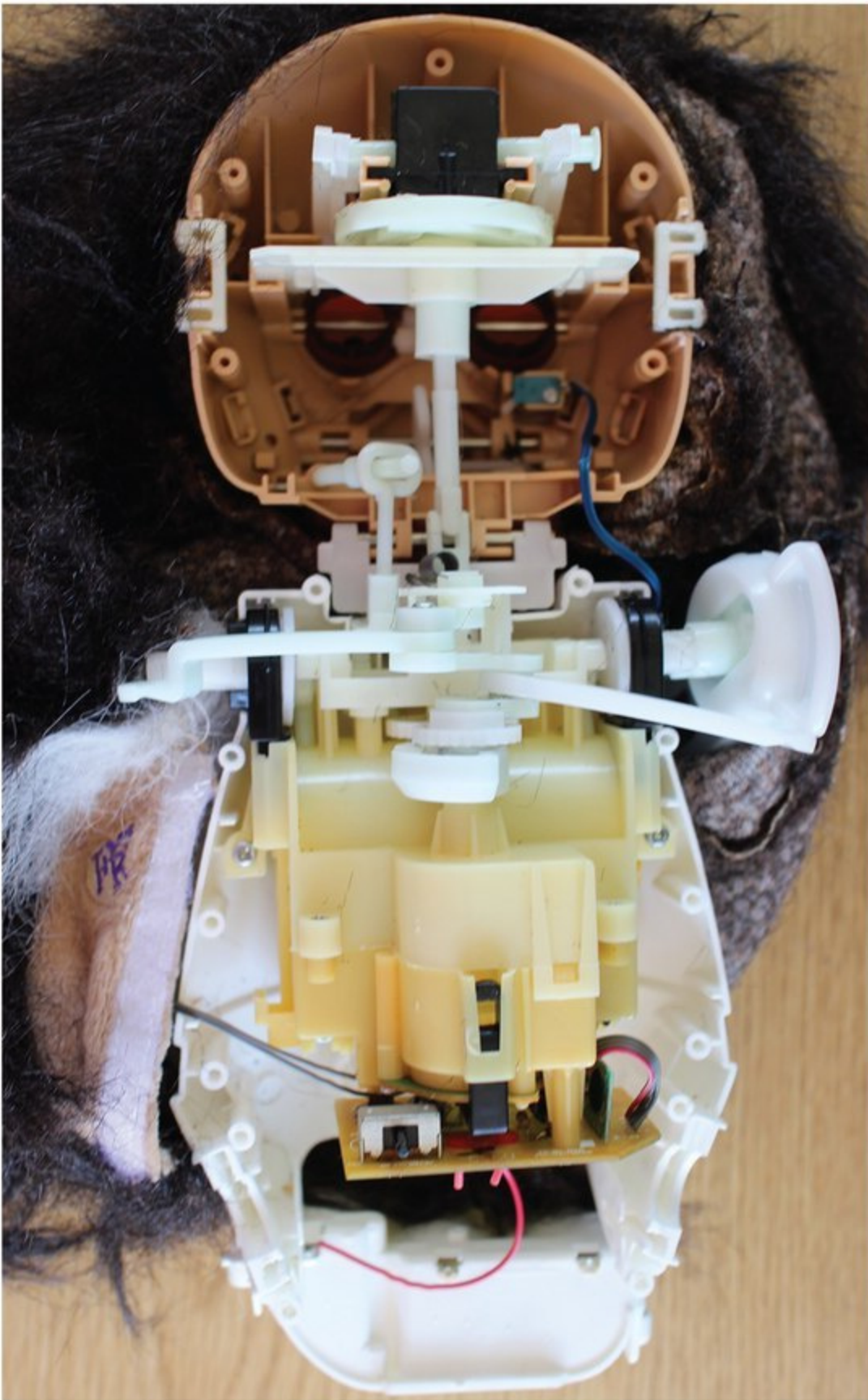


Figure 3. The head interior showing the available space.

to include an onboard audio source.

His circuit was originally designed to be used when a servo was added to a skull (**Figure 3**). This board has been used extensively by those in the Halloween prop building community with many different modifications made.

For my particular application, I first had to determine the best position to mount the servo. The available space was going to be tight — even using a mini servo (**Figure 4**). In order to get the servo to fit in the correct orientation, I had to remove the mounting tabs.

With the tabs removed, I now had to figure out a way to securely attach the servo. I went with one of my favorite methods: using Pro Poxxy 20. This is a two-part epoxy that comes in a tube and is easily combined into a workable putty. You have several minutes to shape it as desired before it hardens, and it provides a very solid connection. You can sand, drill, and tap it once it's cured, and it has a myriad of uses for the prop builder.

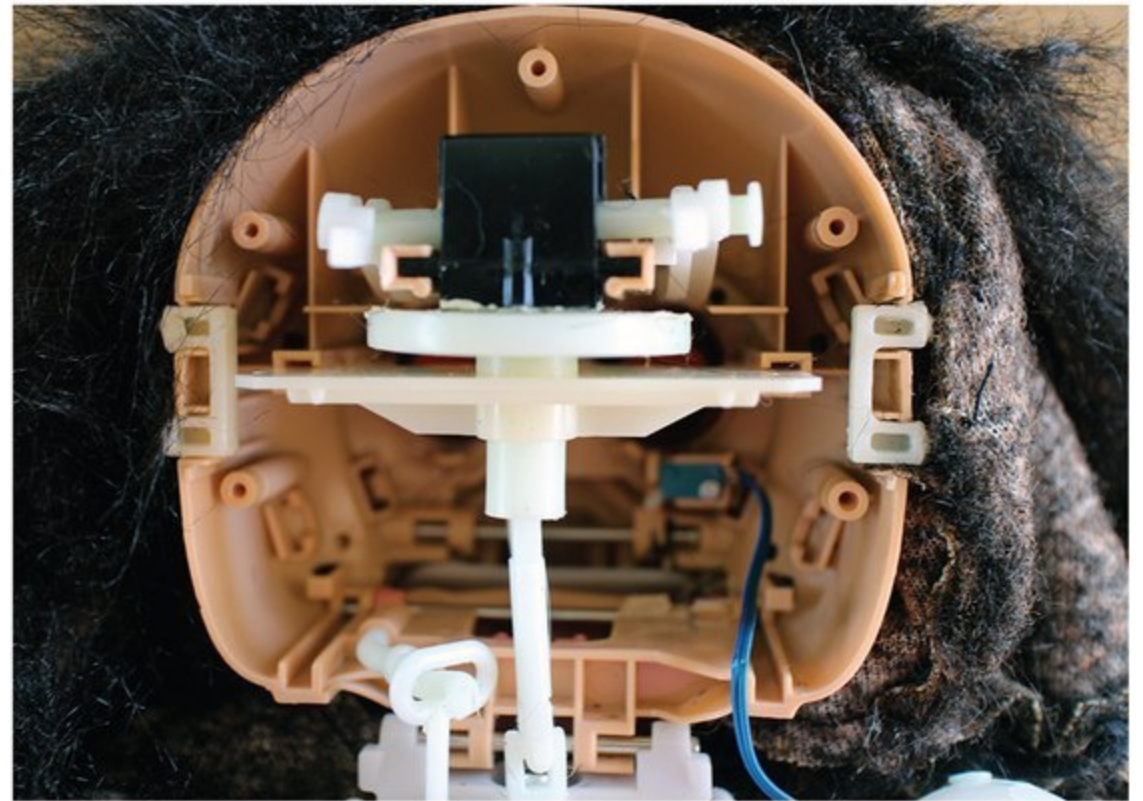


Figure 4. Servo installed and attached to the linkage.

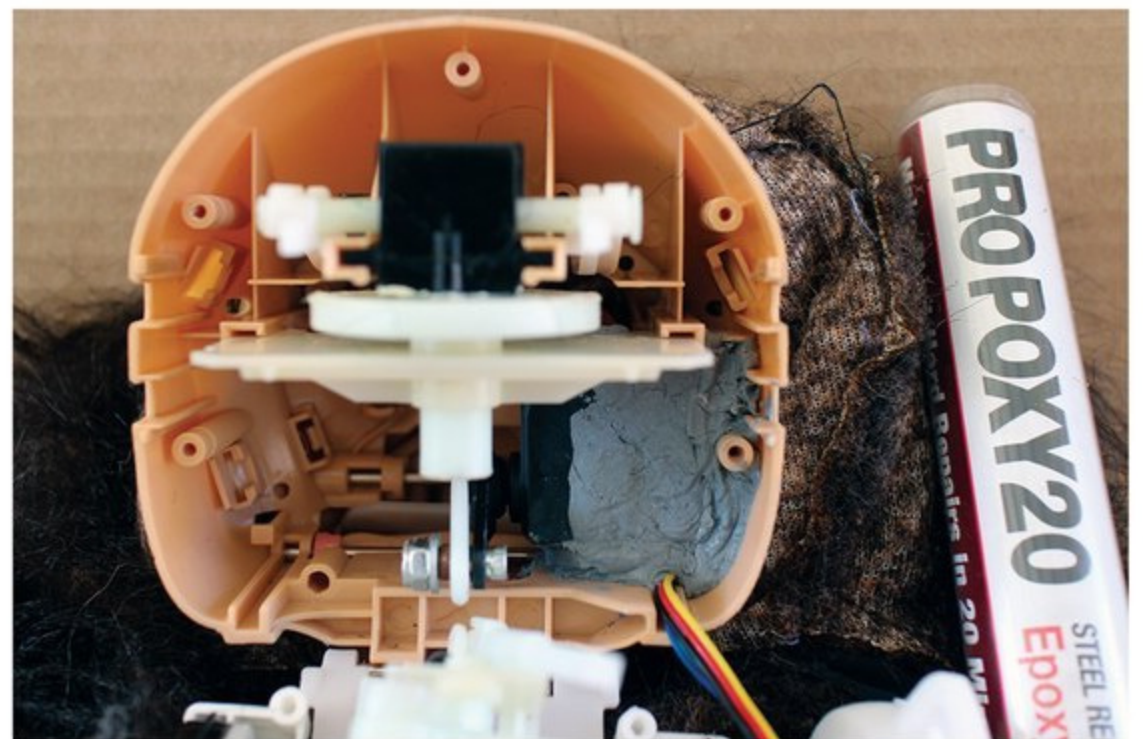


Figure 5. Skulls ready to speak using the Scary Terry design.

Once this was complete, it was a simple matter to connect the servo horn to the existing jaw mechanism. I hooked it up to the controller board and reassembled the monkey. He was quickly talking up a storm!

Let's Get this Skull Moving!

This project utilizes the Lindberg Pirate Skull to which the three-axis kit from Monster Guts is attached. It is suggested that the Hitec 485BB servos be used, but the HS-425BBs seem to work just fine. These kits are so well balanced that the lighter torque servos have no problem providing sufficient force to get great movement.

I've built several different designs to provide the nod, tilt, and rotate motions to a skull, but have never quite gotten the movement that I was after until this kit was introduced. It has solved many of the issues from the other designs and works like a champ!

Several modifications need to be made to the skull before you install the actual three-axis mechanism. Using a

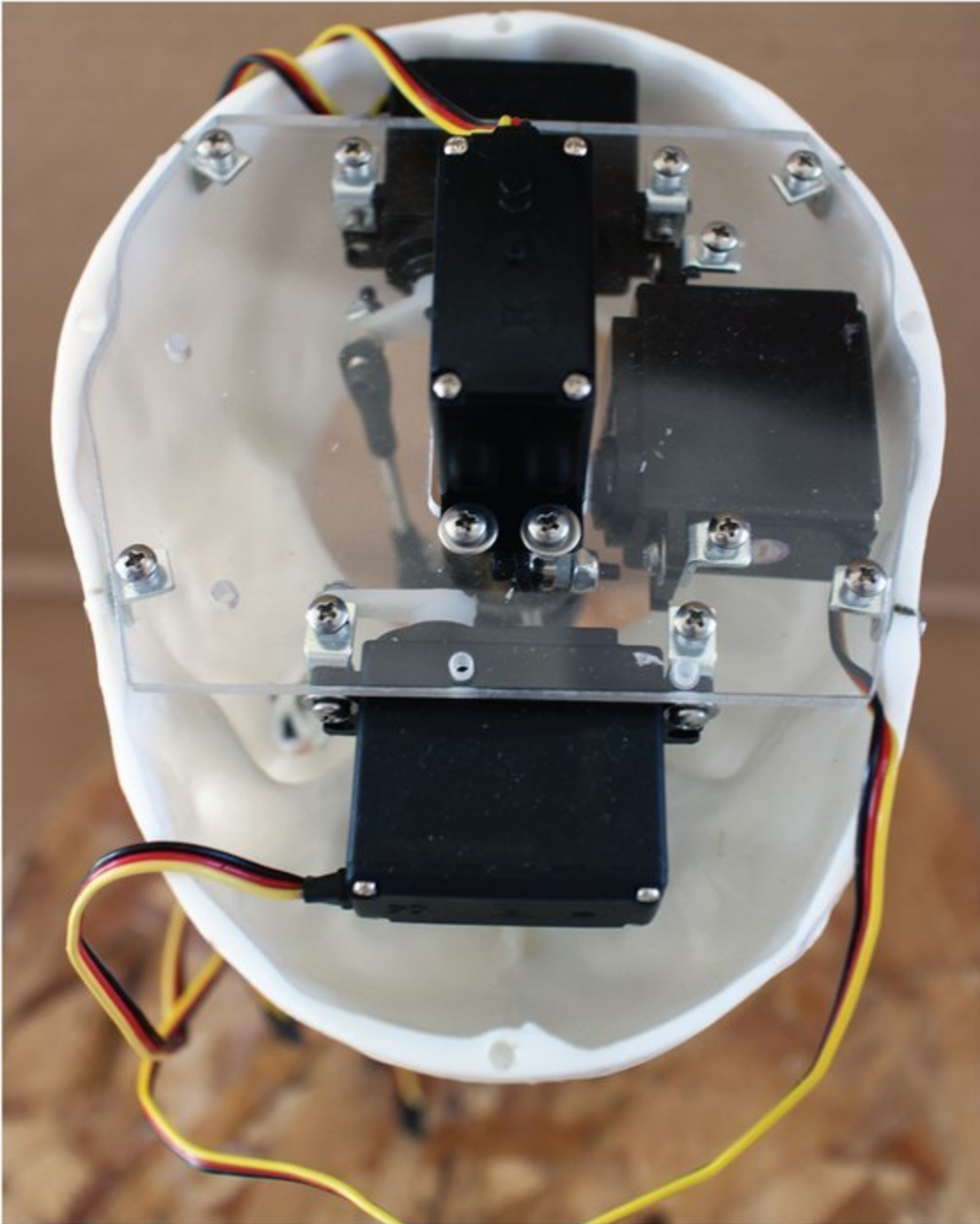


Figure 6. Top view of Monster Guts three-axis mechanism.

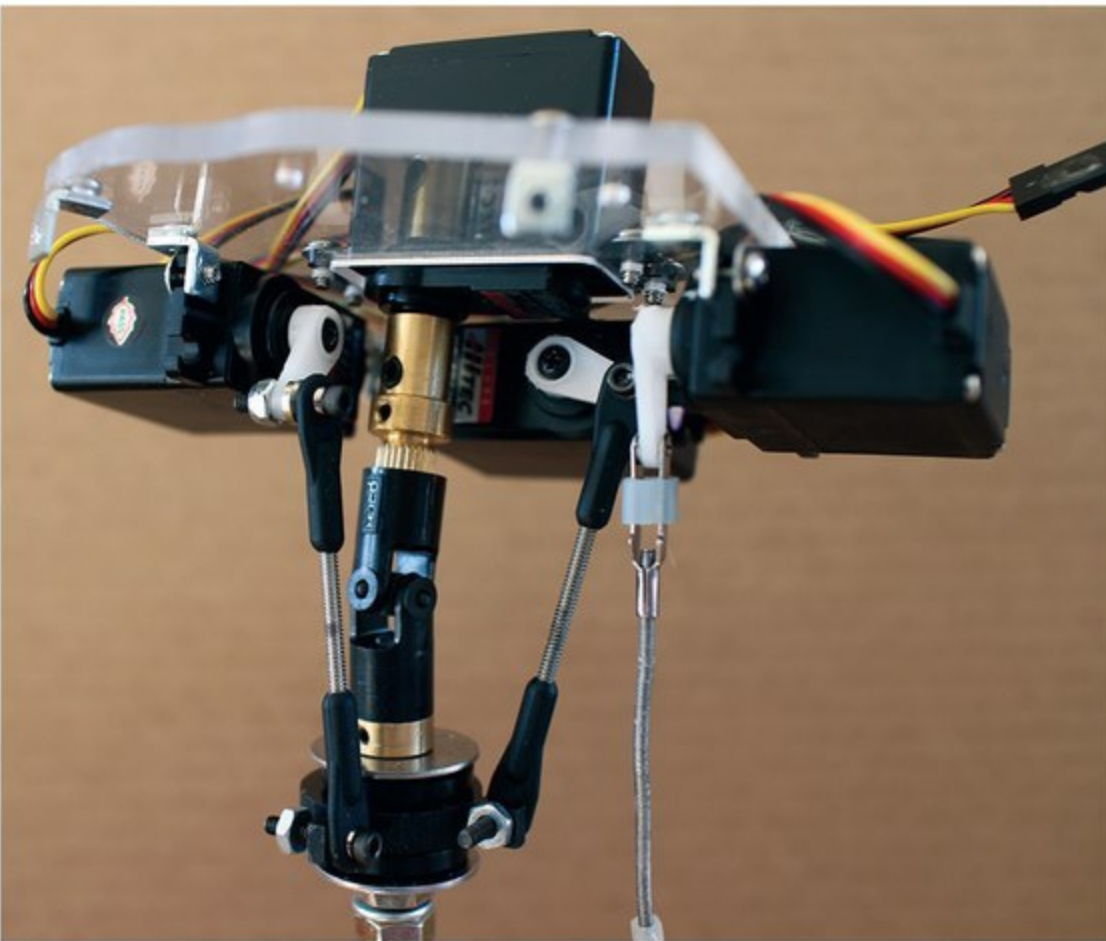


Figure 7. The inner workings.

2" hole saw, the opening in the bottom of the skull is enlarged to allow for the maximum range of motion. Special care must be taken to align your cut properly to avoid removing too much material around the jaw



Figure 8. Kit completed and ready to be mounted to a body.

connection points.

In addition, a hole for the rod that moves the jaw must also be drilled, along with the four mounting holes used to attach the Plexiglas plate. The provided instructions are very specific, so take your time, measure twice, and drill once.

The assembly of the three-axis mechanism is also fairly straightforward, with the provided instructions being very thorough. I recommend that you read them completely before starting the assembly to familiarize yourself with the process. It will go more smoothly once you see it as there are very specific ways the brackets and servos need to be attached (**Figure 6** and **Figure 7**).

If the skull fits with your final character design, you might just need to add a finish to him (**Figure 8**). My usual method uses Minwax Gel Stain, and is quick and easy. Apply it to the skull and then remove any excess with a clean rag. There are many colors to choose from, so pick the one that suits your needs. My favorite is the Honey



Figure 9. Skull sculpted and converted to a new character.



Figure 10. The components for the first prototype.

Maple which is the lightest, but your tastes may vary.

Your project may require a more human looking character than the bare skull provides. We've had good success using the plain skull as a base and then sculpting



Figure 11. Rotate servo mounted in the shoulder plate.

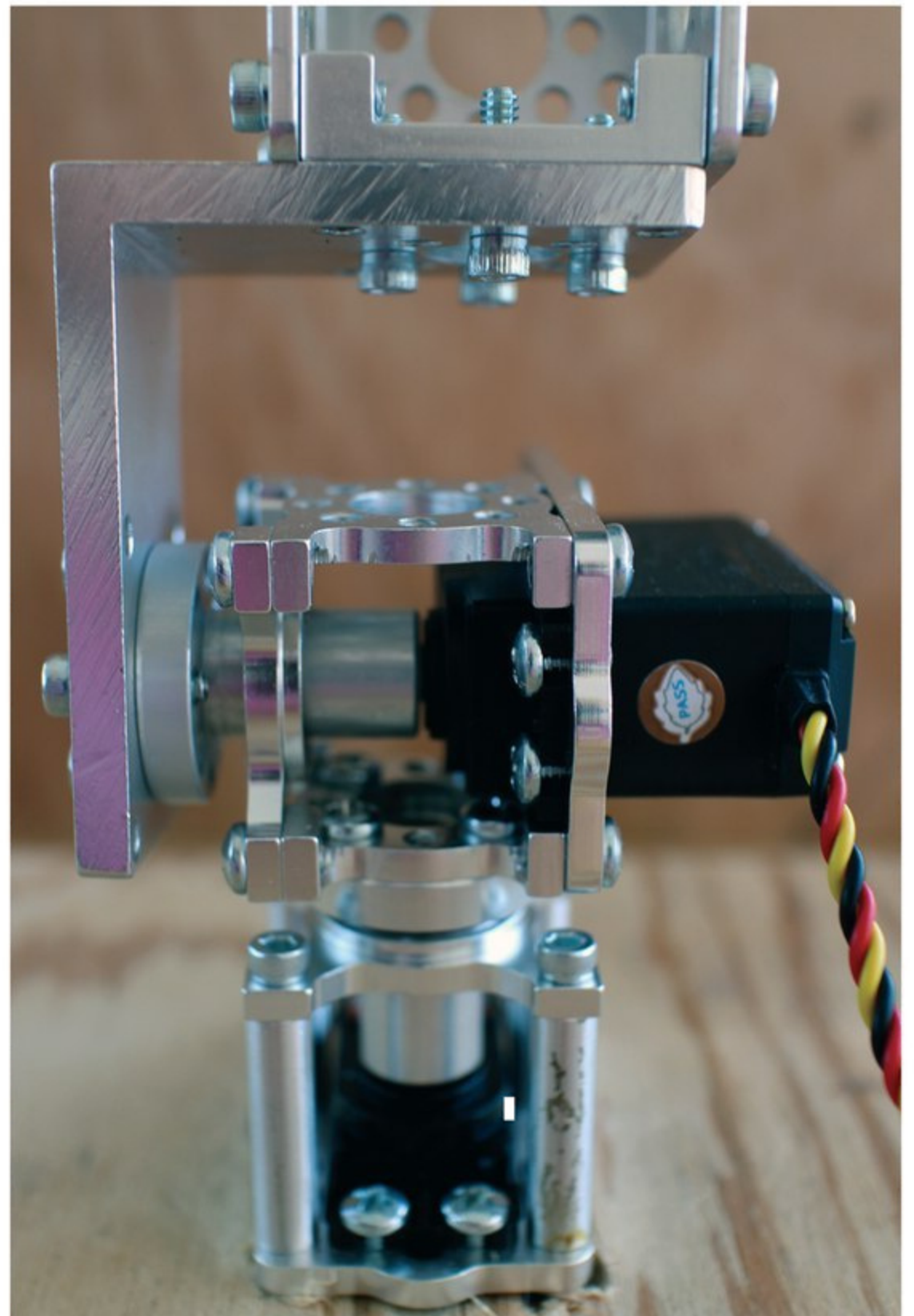


Figure 12. Nod servo added using servo blocks.

on top of it. Robert Risley (a fellow haunter) first tried this on a build we did last year — Pirate Pete. I don't have the space here to properly display his awesome skills but he has documented this process in my forum thread for our Big Red build at **www.hauntforum.com/showthread.php?t=39207**. Make sure to check it out and see how he transforms a skull into whatever character is desired. He

truly is an artist and is developing processes I've never seen done before (**Figure 9**).

This thread is a bit of a bonus for those of you reading this article. It provides the comprehensive progression of the newest addition to our pirate bar scene. If you read it from start to finish, you'll see how I started in one direction and then mid build, totally abandoned all the work I'd done and went in an entirely different direction. I've already got an idea on how to repurpose the original mechanism so it wasn't a total loss, but you'll get an inside look at how some builds end up completely different than the original idea. I plan on doing an article on this build once it is completed with some added surprises, but enjoy the sneak peek for now.

I Want My Mask to be Moving and Talking Too!

The original instructions for this project came from Halloween2Go. (They are included in the first of his three how-to videos.) All are excellent and I recommend that you consider purchasing them.

I really like the concept for this design and have built several that are still in use in my display. However, I was interested in trying to simplify the build process as it does take a significant amount of fabrication to complete.

Once I discovered the Actobotics components from ServoCity, I knew there had to be a way that I could utilize them to make this an easier build. I spent some time on the site and came up with my original concept. I then called and talked to Kyle, one of ServoCity's technical representatives, and refined it until we came up with a workable plan. In no time, the parts arrived on my doorstep and the assembly began (**Figure 10**).

The primary design incorporates two servo blocks which greatly increases the mechanical load the servos are able to handle. They also greatly enhance the available mounting options which allow you to combine them in a number of different configurations. The rotate servo was installed in a cutout in the wood shoulder base to lower the profile (**Figure 11**). I used the Hitec HS-485HB for the rotate and nod servos, but went with the Hitec HS-645MG servo for the tilt. This servo load is considerably more than the other two as it has to lift the entire head back up after it looks down. The HS-645MG has over 50% more torque than the HS-485HBs and is able to handle the additional strain (**Figure 12**). Depending on your mask, you may be able to get by with the less expensive HS-485HB by adding a spring to the back of the assembly (**Figure 13**).

I ended up adding one just to make it easier for the tilt servo and it really made a big difference! This is a great trick that often allows you to get by with a lower torque

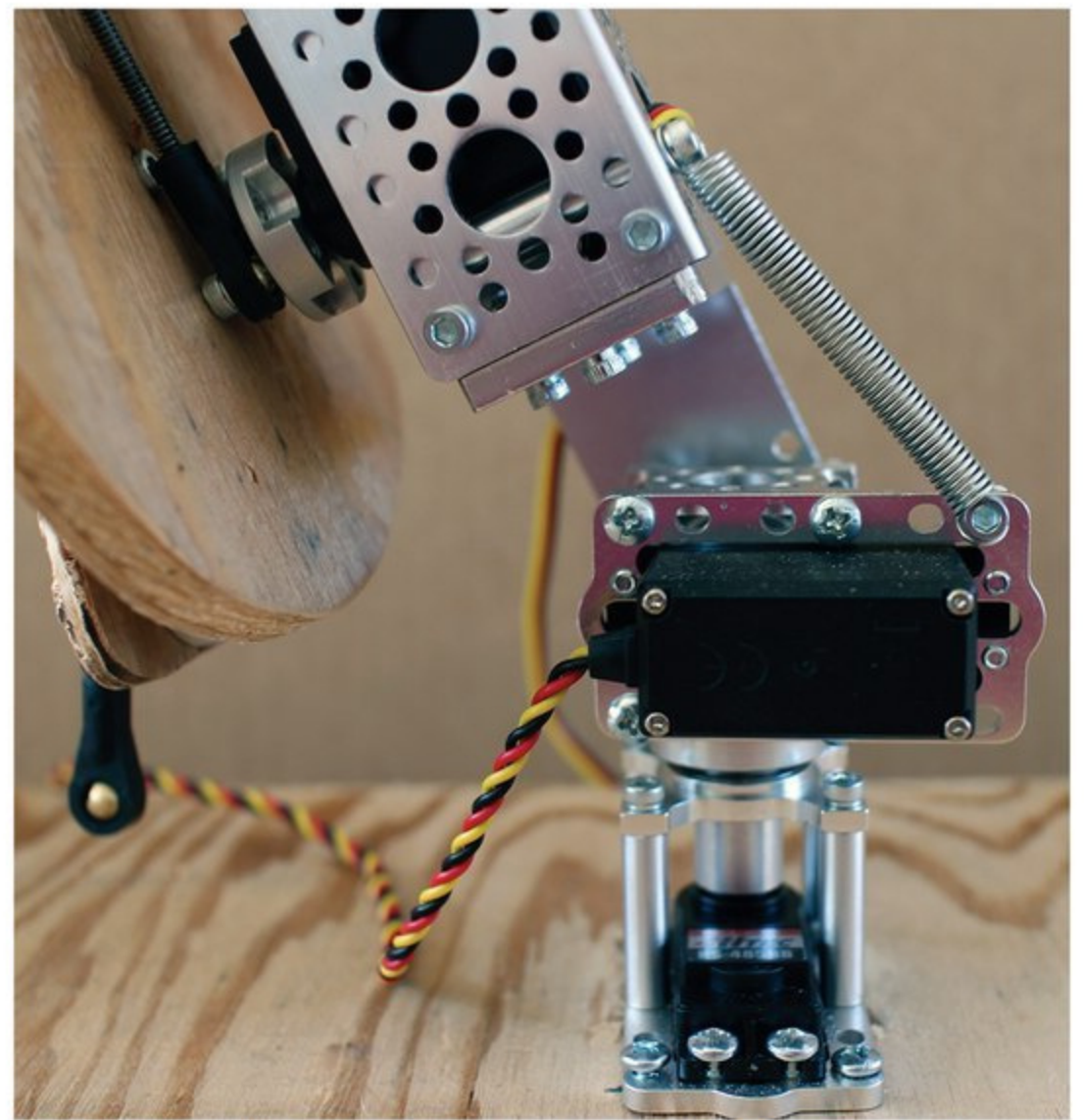


Figure 13. Spring to assist the tilt servo connection.

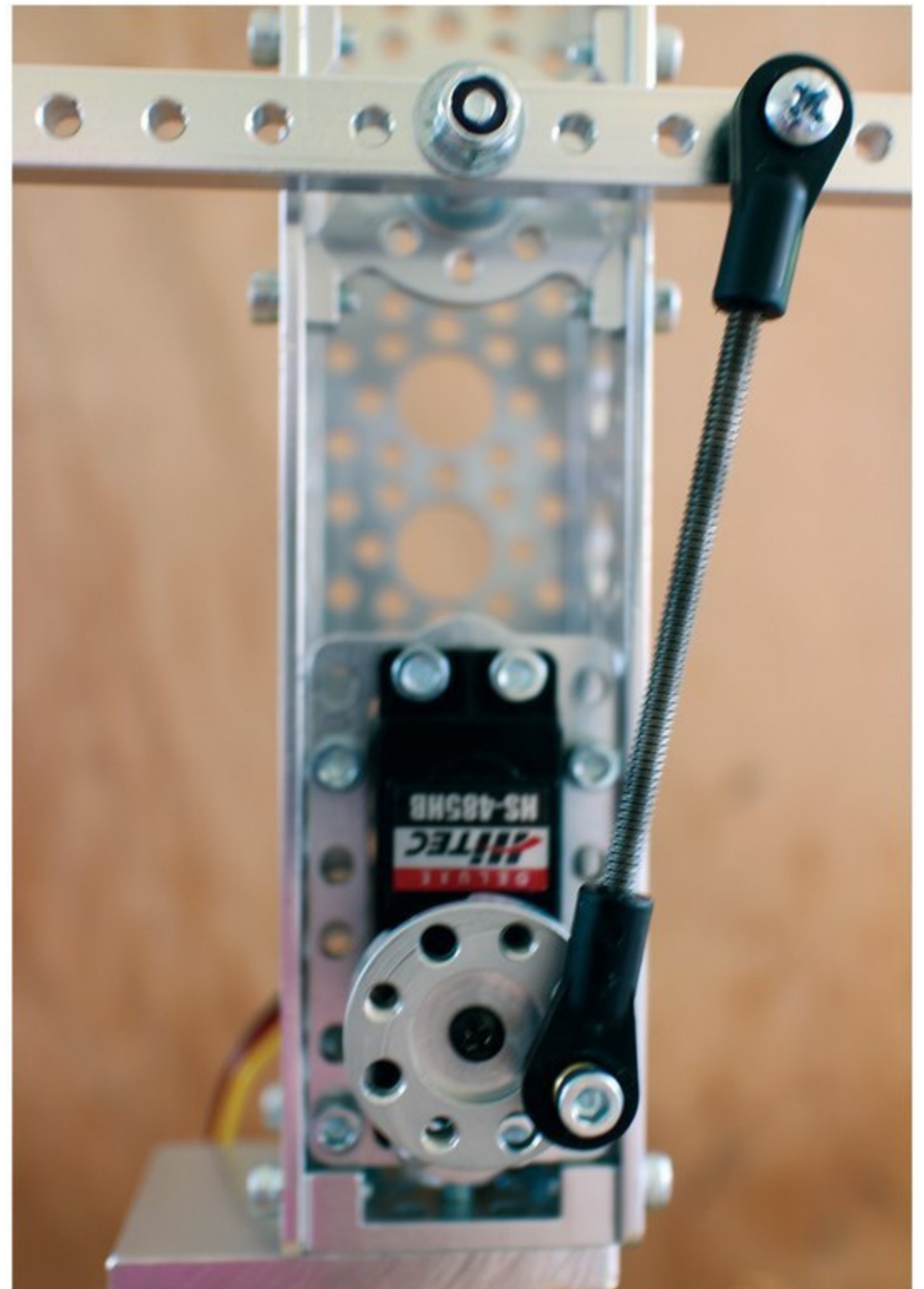


Figure 14. Tilt mechanism detail.

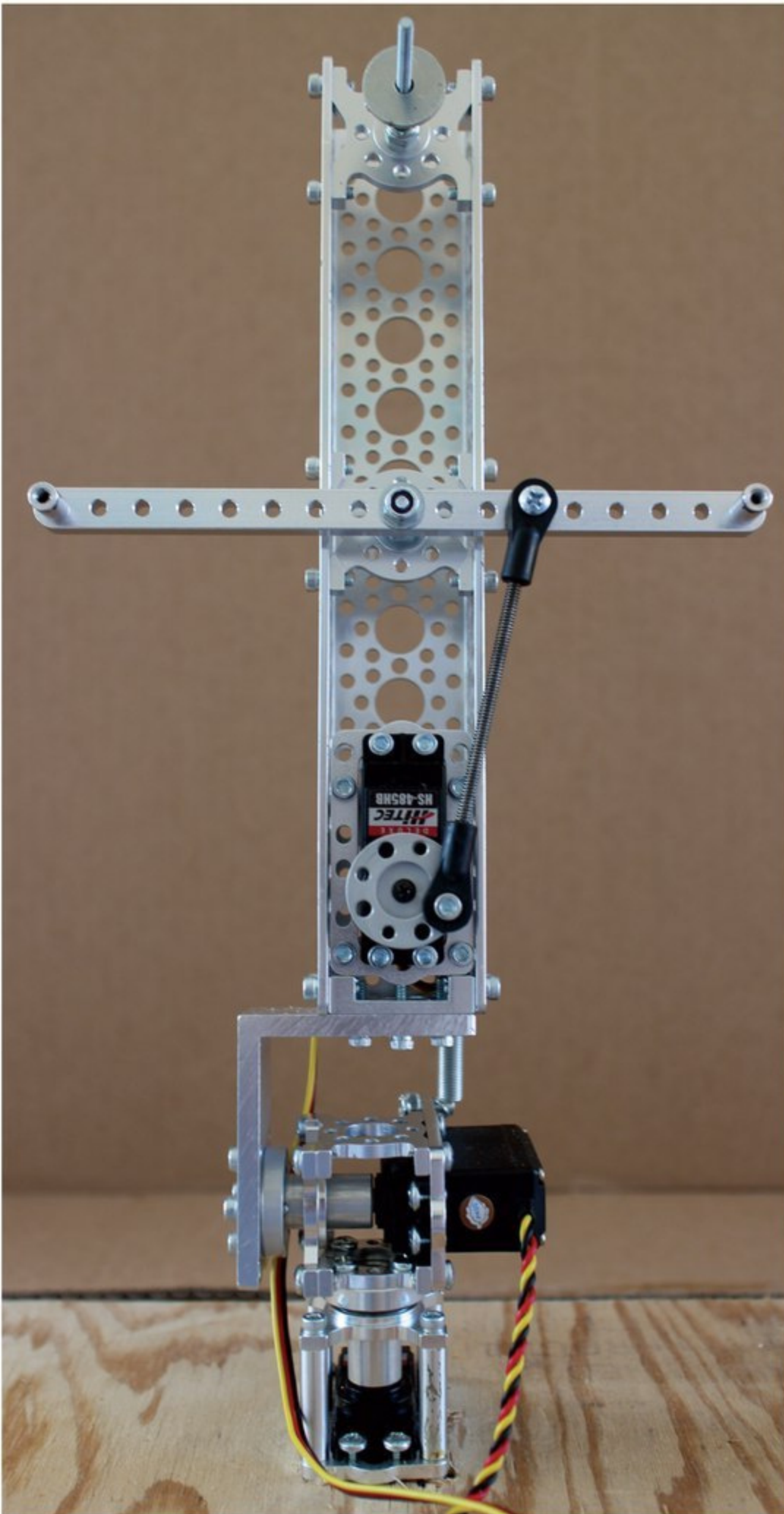


Figure 15. Completed three-axis mechanism.

(and probably less expensive) servo. Sometimes low tech is the way to go!

Tilting the head was accomplished by attaching a 9" piece of channel to the nod servo into which the second HS-425BB servo was installed. One end of a 6" rod was affixed to the servo horn, while the other end was connected to a 7" piece of flat aluminum. The center of this piece was mounted on a swivel joint which allows the head plate to tilt right and left (**Figure 14**). A second hub mount with a bolt that ran in a slot in the wood plate was attached to the channel to provide additional stability.

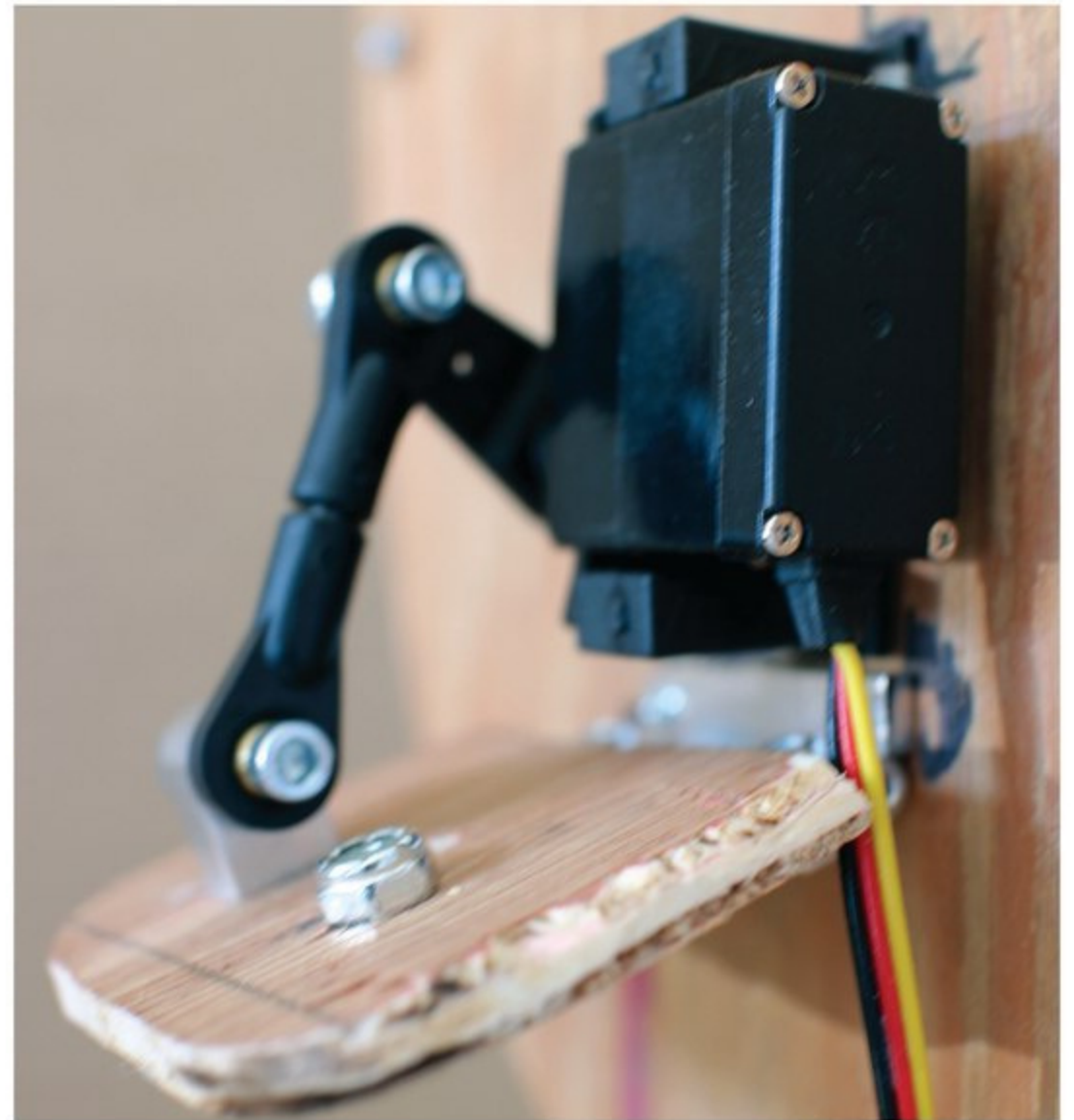


Figure 16. Mini servo to drive the jaw plate.

RESOURCES

www.scary-terry.com/buckyservo/buckyservo.htm

www.scary-terry.com/audioservo/audioservo.htm

www.monsterguts.com/store/product.php?productid=17782&cat=272&page=1

<http://halloween2go.com/volume1p1.htm>

www.servocity.com/html/actobotics1.html

www.halstaff.com

<http://web.audacityteam.org>

The assembly went quickly, and in no time I had a working mechanism ready for the mounting plate to which the mask will ultimately be attached (**Figure 15**). The shape of the mounting plate was first laid out on a foam board where alterations were simple to do. Once I was happy with it, I then transferred it to a 1/4" wood piece.

Additional bulk was added to provide support for the mask by gluing foam scraps to the plate. The foam is easy to carve to the desired shapes necessary to fully support the mask. I'll also add some foam to the top of the shoulder plate to give it the proper shape. The final finishing touch for the mechanism will be the addition of a beige woman's stocking to be placed on the bottom of it to cover the visible portion of the mechanism and to act as the neck.

The mechanism to control the jaw was built using a Hitec HS-225BB mini servo. This servo has more torque than the Hitec HS-425BB in a smaller package. I added a mounting plate to a hinge to which a fitted wood piece



Figure 17. Wood face plate added; just needing some foam.

was added. This was sized to fit the mask and attach to the lower lip. I installed the servo above the hinge. I would have preferred to have it below, but I didn't have sufficient clearance with this particular mask (**Figure 16**). I'm very happy with the results of this design. It came together quickly and only requires a little bit of woodworking with common tools to complete (**Figure 17**). Once some foam was added to provide additional support, the mask was attached and it was finished. I will be inserting some eyes in it, but that's for another day (**Figure 18**).

This build — along with the one for the mask — both utilize my Frankenstein three-axis controller boards. This board drives the jaw in sync with your supplied audio track. It also moves the three head servos — nod, tilt and rotate — in a random fashion (**Figure 19**).

Setup for the board requires you to simply download the supplied program, open it in the free program editor

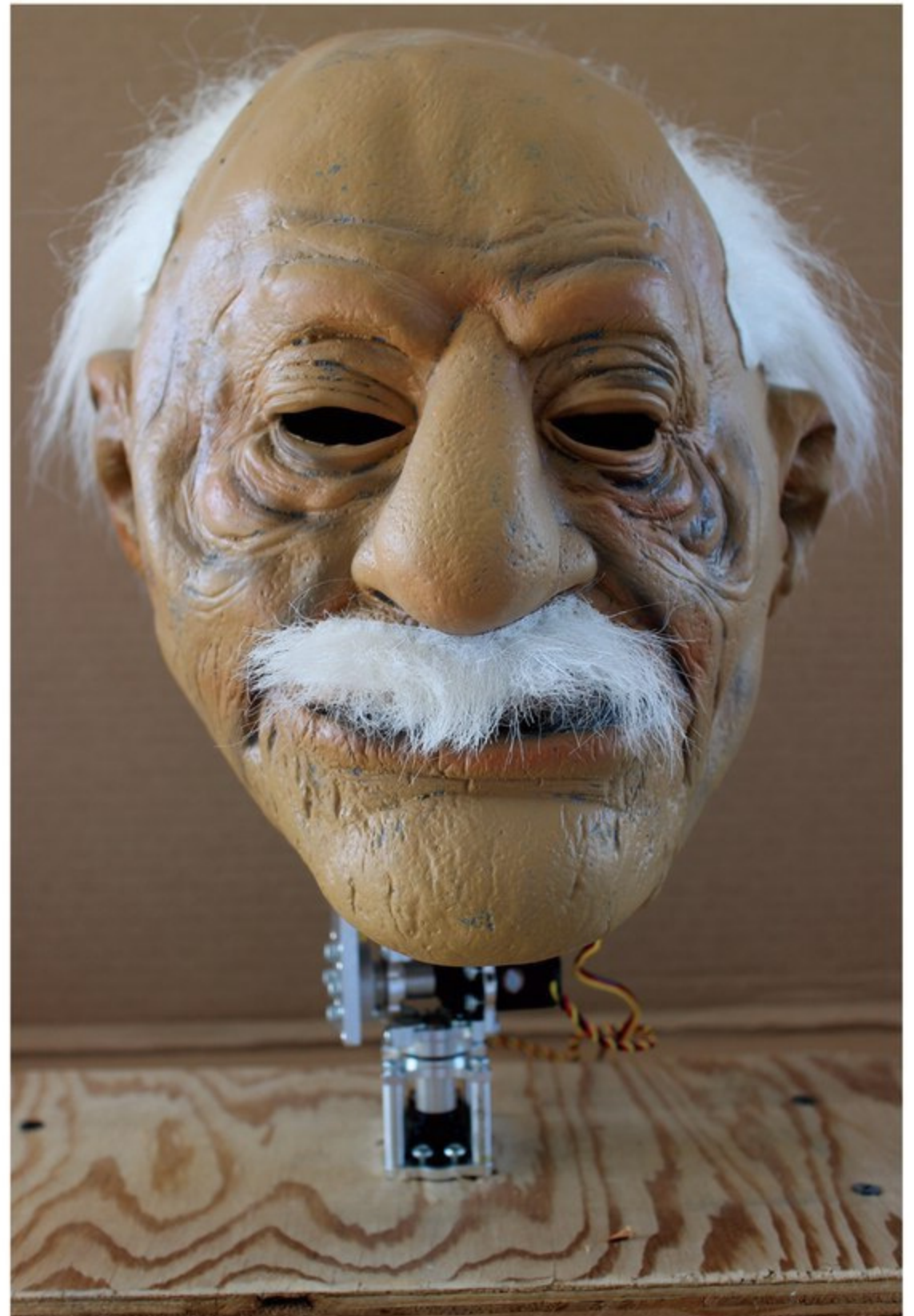


Figure 18. Mechanism completed with mask added.

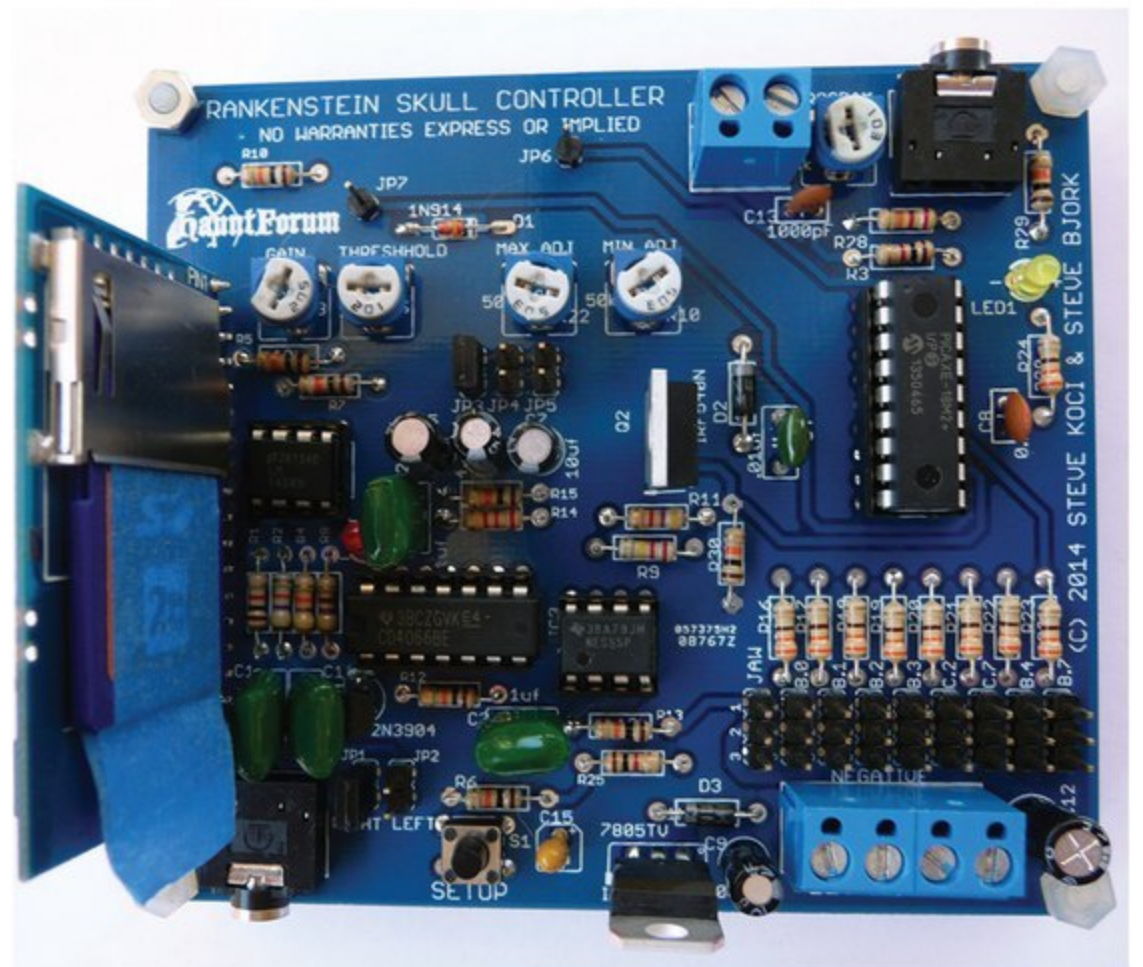


Figure 19. Completed Frankenstein controller.

provided by PICAXE, and load it to the chip.

The entire setup process for adjusting the servo limits, trigger type, retrigger delay, number of both the ambient and trigger audio tracks, and adjusting the jaw servo are done by following audio prompts that have you adjust a potentiometer to the required positions and then push a button to set them. The entire setup process takes less than 15 minutes.

The audio tracks are stored on an SD card with the ability to have up to four ambient tracks and 10 trigger tracks.

We designed the board to be easy to set up and to not require a computer once the original program is installed on the chip.

This board is the culmination of several years of learning and experimenting. The desire to create a board to control a three-axis skull is what started me on the path to learn how to design and build my own custom printed circuit boards.

Now with that experience under my belt, I'm usually able to put together a board to meet most of my prop building requirements.

ACTOBOTICS BUILD PARTS LIST

Quad hub mount C	545360	1
1.125 Standoffs	633132	1
6/32 Locknuts	632142	1
90° Quad hub mount B	545424	2
.500 Standoffs	633122	1
Swivel hinge	585644	1
Flat single bracket	585468	1
Attachment blocks	585403	1
9" Channel	585450	1
190° Hub mount A	585494	1
Linkages	585432	1
Center hole adapters	633118	1
Hub horn	525130	1
Aluminum beam	585416	1
1" Standoffs	633130	1
Servoblocks	637110	2
HS-485BB Servos	33485S	2
HS-645MG Servo	32645S	1
HS-225BB Servo	31225S	1
6/32 Threaded rod	98847A007	1
.3125 6/32 Screws	632108	1
.500 6/32 Screws	632114	1
.625 6/32 Screws	632118	1
Servo mount	SVM275-115	1
Servo plate	575112	1
6/32 Screws from hardware store in 1.25" & 1.75" lengths		
Spring		

Time to Add an Audio Track

To achieve the most realistic movement of the jaw in relation to the audio, we use a free audio editing program called Audacity to modify the tracks. We replace the audio of the right channel with a tone track which is what actually drives the jaw circuit. Using this method, the jaw will only respond to the desired talking portions of the audio clip and allows us to also have background music or sound effects that the jaw ignores. The entire clip remains on the left channel and goes to your speakers.

You can record your own voice with a custom track and then adjust it to fit your needs through the program.

I suggest you use quality speakers that take full advantage of the very good audio provided by the Tenda boards. I mount mine in the chest of the figure as it makes your character more believable and helps cover any servo noise.

Now it is Your Turn!

This can be a challenging undertaking, but the rewards are worth the effort. The confidence gained by completing one of these projects will certainly encourage you to take on more complex undertakings.

Nothing adds to the interest or believability of a character more than for you to give it the gift of speech. Adding the three-axis movements is the icing on the cake.

So, pick a design and start building! But be warned ... once your characters start talking, you'll want them all to have the gift of gab! **SV**

WORLD'S MOST VERSATILE CIRCUIT BOARD HOLDERS

Model 324

Our Circuit Board Holders add versatility & precision to your DIY electronics project. Solder, assemble & organize with ease.

VISIT US ON



MONTHLY CONTEST
 Visit us on Facebook® to post a photo of your creative PanaVise project for a chance to win a PanaVise prize package.

PANAVISE®
 Innovative Holding Solutions

7540 Colbert Drive • Reno • Nevada 89511 | (800) 759-7535 | www.PanaVise.com

Model 201

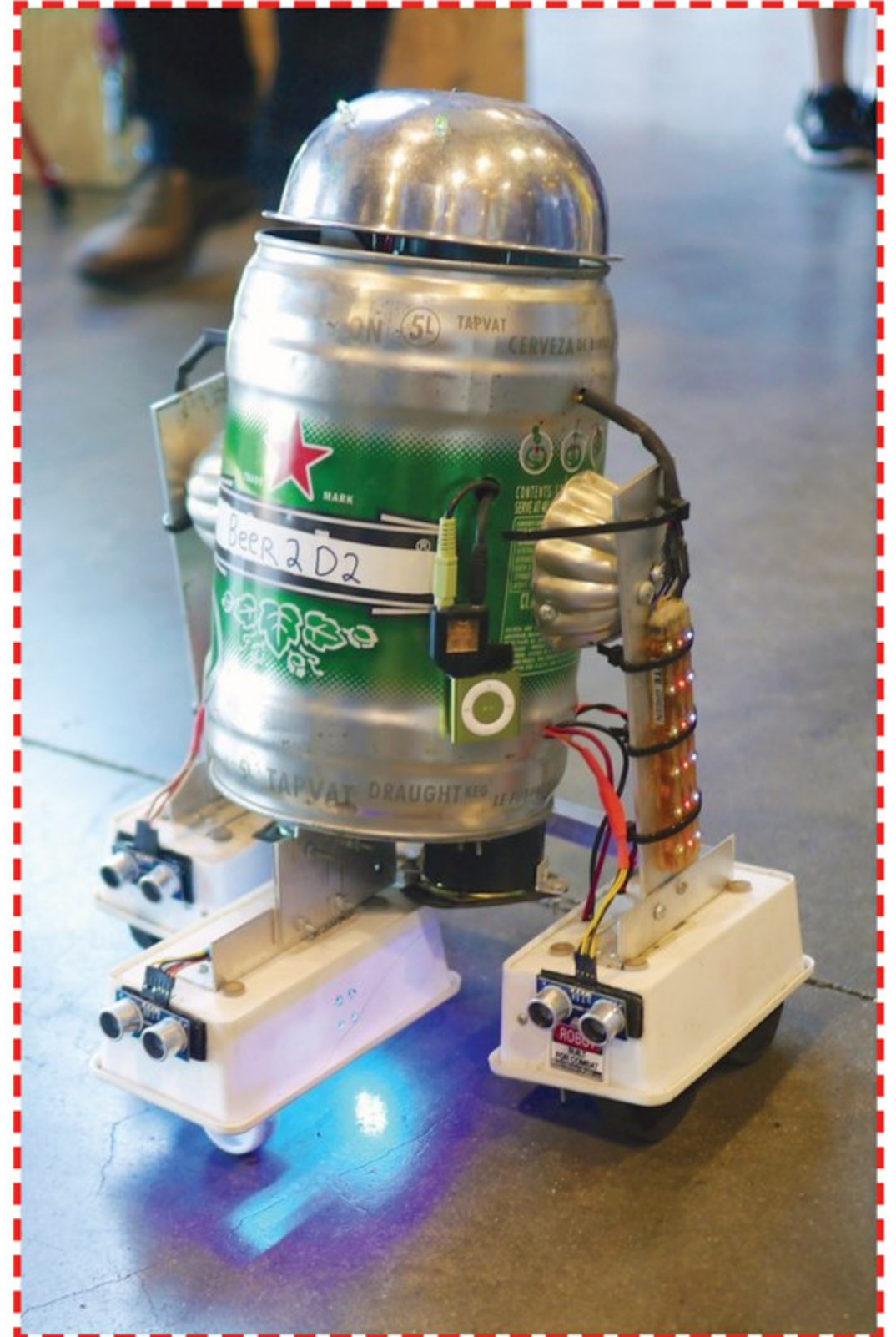


★ Beer2D2 ★

A Robot in a Can

By Steven Nelson

There are many ways to begin a new robotic project. You can start with one of the many robotic kits that are available from vendors such as Parallax, SparkFun, Adafruit, etc. You can do a lot of searching for raw materials, electronics, motors, controllers, and wheels, and then ponder what to do with them. A more modern approach may involve finding the parts you wish to use and then beginning your design by laying everything out using a CAD program, then eventually a milling machine, lathe, or even a 3D printer. Then, of course, there is the old school classic design method that involves collecting the various parts and sketching out your rough ideas with a pencil and a piece of paper. Personally, I usually start with that. In talking to my friend, Camp Peavy (Neato Robotics), he told me, "You can build a robot out of anything and do it very quickly if you use a lot of that magical material called Obtainium." (Obtainium being any parts or materials you can "obtain" and re-purpose.) Of course, this may turn you into a bit of a pack rat since almost everything becomes re-usable again ...



I had an Itch to Scratch. I Needed to Build Something.

About five years ago, I was working at two different jobs in two different cities and had very little free time for my projects. Having a job and working is great, but for me I never stop thinking about making some new robot or toy to play with. In about mid February of 2010, I was aware of the upcoming RoboGames event that would be held in April. I got to looking at my "Obtainium" pile and noticed an empty 1.5 liter Heineken keg that I hadn't thrown out yet. Boy, I thought, that shape sure reminds me of a world famous astro-droid from another galaxy.

I grabbed that can, an old set of amplified computer speakers, and a 75 MHz Novak radio receiver. I then started looking for some tires and drive motors. I didn't have any that weren't already in use. I looked up on a shelf and saw my little 4WD robot called EVA. For a moment, I thought, NO! You can't strip EVA for parts! EVA was actually a training aid I made so I could learn how to build and program autonomous robots using the Parallax BASIC Stamp 2 and various sensors. Plus, she had a pan and tilt wireless camera, a tele-operated mode, and a gripping arm that could retrieve 12 ounce drinks/cans from the refrigerator. It was a really hard decision to borrow her electronics and motors, so with a slightly heavy heart I decided to go for it.



Beer2D2 parts.

Building Beer2D2 from Random Bits of Stuff

Once I knew I had some of the major components for the robot's drivetrain, I had to come up with a basic design for the rest of the body. Out came the tape measure plus the paper and pencil. I measured the Heineken keg's length and width, and guesstimated its inner volume. I measured

EVA's tread width and wheelbase including the tire diameters, plus I weighed her. She weighed six lbs. Since I knew EVA would run for over an hour with a 2,000 mAh 9.6 VDC NiCad battery pack, I figured the new robot should do so as well. At this point, I needed to actually buy/find some new parts.

My first stop was the Budget Robot Builder's Emporium (Goodwill store). The first things I found in the used kitchenware section were some tin cupcake molds. I thought, "Ah, these will make excellent shoulders." Then, I had a problem. I needed a domed head for the robot. Since I had measured the top of the Heineken can and knew it was 5-1/2" in diameter, I began digging through dishes, glassware, cups, saucers, plastic kid's toys, and nothing fit! I was almost ready to give up when I saw a stack of stainless steel mixing bowls on the bottom shelf.

I got down on my hands and knees, pulled the stack of bowls out, and right there two bowls down from the top was a stainless steel bowl that was the perfect size. I quite loudly exclaimed "YES!" Several of the women that were also shopping looked at me like I was crazy. (Little did they know, they were probably right.)

I meekly apologized for my sudden outburst and greedily grabbed my rare treasure. I grabbed a few other bowls but I didn't end up using them since none fit as well. The next thing I needed was something to mount the drivetrain into.

Astro-droids have these rectangular wheel parts and I needed three of them. Again, I searched through lots of cake pans and metal dishes by placing the servomotors and 3" tires in them, and found nothing. So, I took the treasures I had already found, paid \$4 for them, and went to Walmart to search for more ... and found nothing.

At this point, I headed toward Home Depot to get some sheet metal to make my own drivetrain enclosures. Then, I saw a K-Mart store and whipped into the parking lot. I showed the cashier my motors and wheels, and asked where their kitchen appliance section was — triggering yet another strange look.

In the kitchen section, I was very lucky to find the only three plastic butter dishes that my motors and wheels would fit in. I believe they were from the Martha Stewart collection, and cost me about \$11. Finally, after that purchase, I went to Home Depot and bought three feet of 3" x 1/8" aluminum flat bar, and three feet of 1-1/4" angle aluminum stock — plus a bunch of 1" long number six and



Beer2D2's dome motivator.



number four bolts and nuts, and a roll of self adhesive Velcro™. These parts cost around \$45. (I bought extra.)

A Box of Happiness

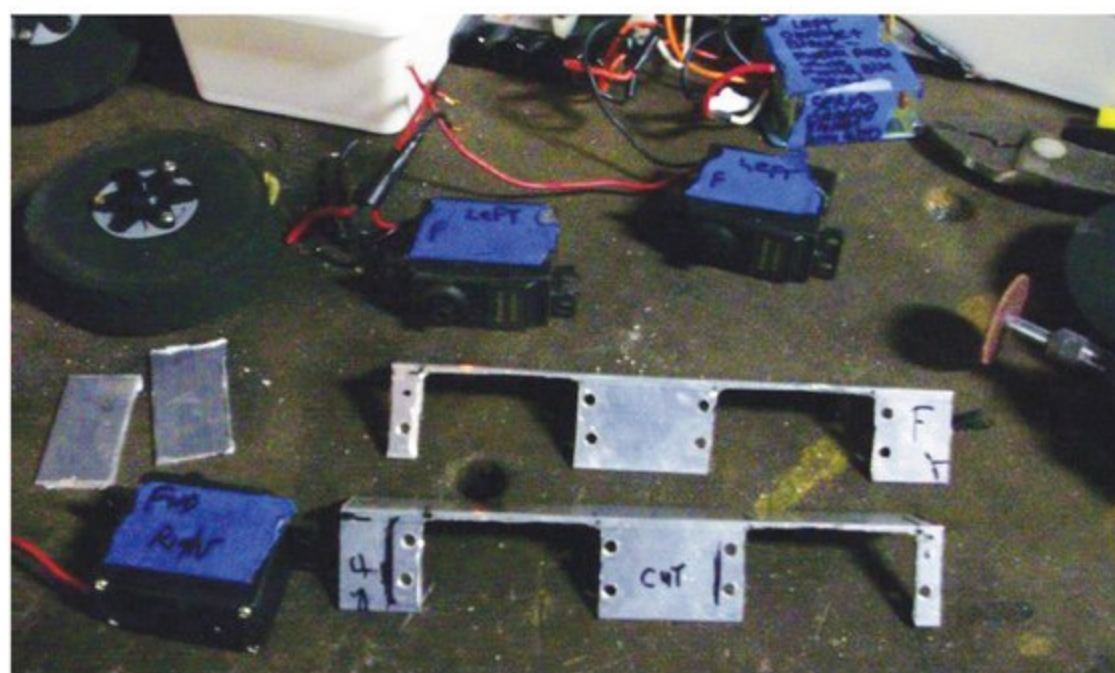
I put all of the parts into a box and hauled them to my second job. I began the project in their shop late at night after work. I needed to cut open the top of the Heineken can, and knew that the can had been pressurized. So, I took a hammer and center-punched a small hole in the top. I was gifted with a whoosh of gas and the unpleasant smell of very old beer residue (eew).

I wanted to leave about 3/4" in the top of the can's metal, so I could mount the domed head. I used a piece of stiff solid core wire, a sheet rock screw, and a Sharpie™ pen. I put the sheet rock screw in the center hole where the can's dispenser valve used to be, and made a loop in the wire to hold the Sharpie. I then used the wire/guide (pulled tight) to trace out a circle on the top of the can. Using a Dremel rotary tool and a cut-off wheel, I cut out the circle, removed it, and the exhausted internal gas cartridge. I gave the can a thorough bath inside and out in hot soapy water to remove the smelly old beer residue. I then proceeded to totally disassemble EVA, so I could get her two Vantec RET 411 motor speed controllers, battery packs, and power switches, and lay them out on the table.

There were several test fittings done to see how to get all of the parts inside of the can, while pondering how to wire Beer2D2 and make it easy to do maintenance and battery charging. In the initial design, I placed the speed controllers and the battery packs in the very bottom of the can to keep the robot's center of gravity low so it would be semi stable. I placed the entire speaker system on top, and it all fit. Time being

a factor, I figured that was good enough for now. I had to build the drivetrain/chassis. This part of the build I actually did over a couple of nights.

Beer2D2's drivetrain uses four Futaba S3004 servos hacked for full rotation, with all of the original electronics removed. They are mounted with 4-40 machine screws to a piece of angled aluminum that was notched using a Dremel to fit



Beer2D2's motor frame.

the motor's case. Each set of motors is wired in parallel and connects to one of the Vantec RET 411 speed controllers. The motors are fed 9.6 VDC from a 2,000 mAh NiCad battery pack. Actually, 9.6 VDC is a good over-voltage level for stock hobby servos with nylon gears. If you use a higher voltage, you should use servos with metal gears.

It was a bit tricky fitting the motors and 3" Dave Brown Lite Flite wheels inside of the butter dishes/motor housings. All of the holes I drilled were 1/8" and the drill bit liked to grab the plastic as it went through. As usual, I ended up bleeding a bit from the cuts acquired from reaching into a metal can.

For the front wheel housing, I made a bracket with some 1-1/4" angle aluminum and a long through to act as the axle, to support another non-driven 3" Lite Flite wheel. Eventually, I would regret this.

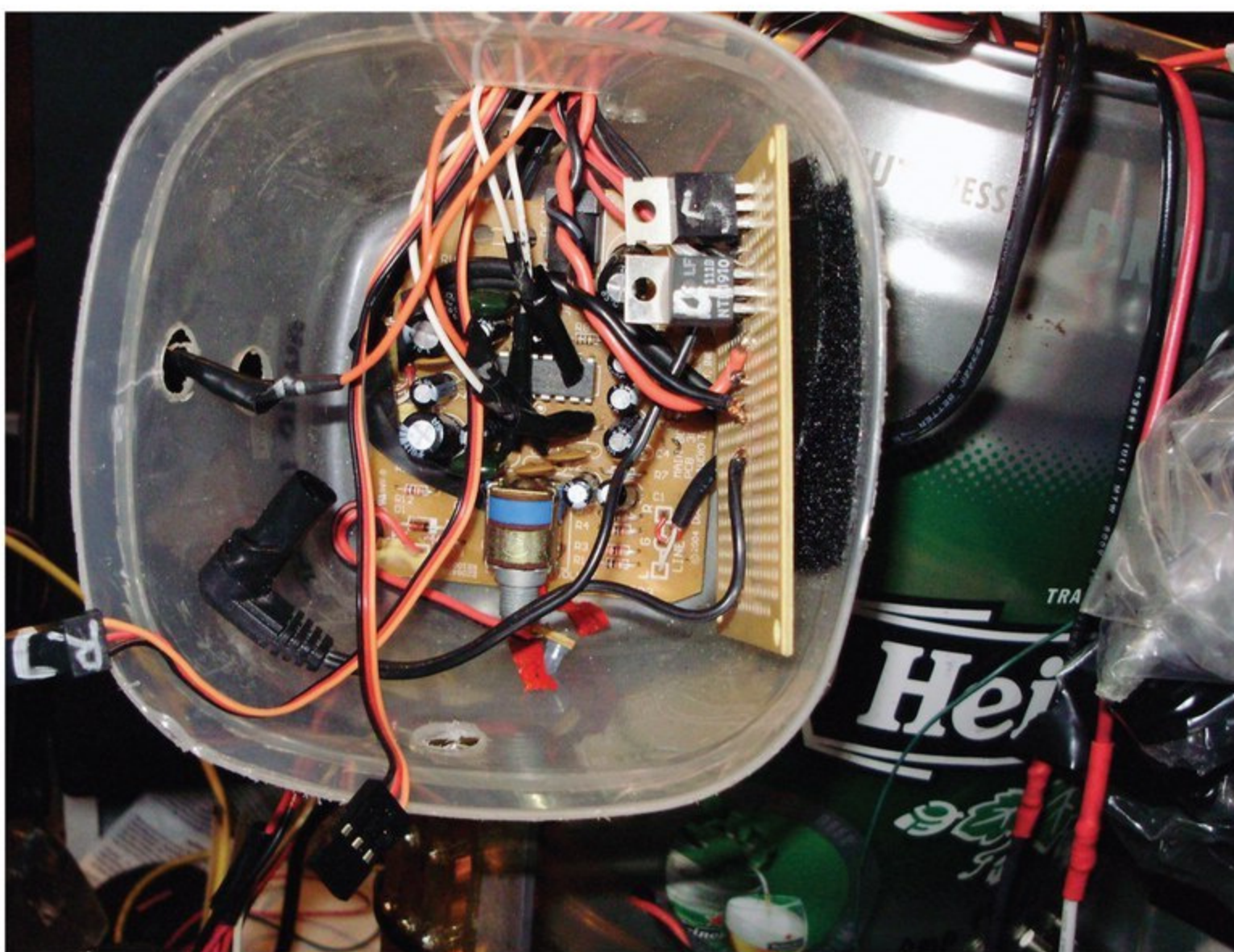
After I had assembled the housings and tested the



Beer2D2 and my handy Dremel.



Beer2D2's feeding time.



Amplifier and power supply.

drive motors with a battery, it was time to figure out how to attach them to the Heineken can.

Trying to mount things evenly on a round can gave me a few things to think about. How was I going to measure all of the places I needed to drill the holes? After some pondering, I used a pocket tape measure that was very flexible. I could wrap it around the can to take measurements and mark the places for the holes. I also wrapped some blue painter's tape around the can to give me some straight edges to work from. The printed decal on

the Heineken can has a star in the center of it, so I used its center line to work from, as well.

Once I found the positions for the shoulders, I attached them with a couple of number 8 sheet metal screws. One thing I learned about Heineken cans is they are incredibly thin walled and not easy to drill through without tearing. The drive housings are attached to 1-1/4" angle aluminum that bolts to the 3" aluminum flat bar that acts as Beer2's legs. They attached to the shoulders using several number 8 sheet metal screws. The front wheel housing is attached using 1-1/4" angle aluminum connected to another piece of 3" flat bar, connected to yet another piece of 1-1/4" angle aluminum.

Test Run and the Typical Disaster

Once I had it all bolted together, I put the batteries, speakers, and electronics in and set it on the floor. It fell over. Ah dang, I had a center of gravity problem. I went back and watched the real R2D2 in the movies, and noticed that he actually leans back a bit when he is rolling around. Ah ha! Better building through watching sci-fi movies! I had to reposition/tilt Beer2D2's main body back until he became stable.

After all of that, I finished the four battery power switches, motor, speed controllers, radio receiver wiring, and took him for an R/C test drive. Yay, he works! It was now time to give Beer2D2 a little personality, so I connected his 12 VDC 2,000 mAh battery pack to a switch mounted in the can, connected the amplified speakers, and added an Apple iPod Shuffle for an audio source.

I also mounted a stock servo to another piece of 1" angled aluminum and found the inside center of the stainless steel mixing bowl by carefully balancing it on a Sharpie pen to mark the spot where I drilled the 1/8" hole for the servo motor's center shaft screw.



A Beer2D2 fan.

Web Links for Beer2D2

Jameco LED color organ — now in kit form!
www.jameco.com/Jameco/workshop/diy/ledcolororgan.htm
 !?CID=organ

A much simpler and smaller LED color organ from *Make Magazine*
 The Easy LED Color Organ
<http://makezine.com/projects/easy-led-color-organ/>

Vantec speed controllers
www.vantec.com/acatalog/spdcat.html



I also drilled a couple of holes and mounted a couple of green LEDs in the bowl to act as his eyes. Once the head was mounted and rotating about 90 degrees under radio control, he was pretty much done.

The First Adventure at RoboGames

Beer2D2 and I went to RoboGames for the first time in 2010. We actually had a lot of fun hanging out with the ArtBots and the folks from the Homebrew Robotics Club. To my surprise, the kids really loved the little guy. We drove all over that event. He even got to meet the full-sized R2D2 projects from several talented builders.

Then, I started noticing several problems. The fixed front wheel was causing a lot of drag during turns and was also trying to tear itself off of its rim sideways. Another major problem was happening with the drivetrain legs and their attachment to the tin cupcake holders/shoulders. The legs began shifting and flexing, and starting to tear the mounting screws out of the cupcake molds and the main Heineken can. Not good! I stopped driving him on the second day of that event and just put him on display.

Time for Some Upgrades, then Off to the Bay Area Maker Faire

When we got home from that event, I accessed the damage. I strengthened the shoulder mount, added a piece of aluminum channel between the drivetrain housings, and bolted that channel to the can with some more 1" angled aluminum. That made the drivetrain much stronger, and there was no more flexing.

I removed the fixed front wheel and replaced it with a 1" self-aligning caster wheel. That took care of the tire drag and helped him turn easier. Now, for a serious road test!

In May 2010, I took him to the Maker Faire, along with several other projects. He was working a lot better.

Even More Upgrades

After that Maker Faire, I wanted his sound system to work better. The speakers were really muffled from being inside of the tin can. So, I took him apart and disassembled the amplified computer speakers.

Once the speakers were free from their housing, I



Beer2D2 at RoboGames 2013.



Beer2D2 and R2.

found that they were about 1-1/2" in diameter. I took one to Home Depot and found a PVC pipe coupling that they would fit in, so I bought two. I mounted them to the bottom of Beer2's can, along with some of their original speaker grill. I put the amplifier inside of a small Rubbermaid™ container that I found at Walmart and put it all back in the can. That really improved his sound.

One of the other projects I had demoed at that particular Faire was an audio controlled LED color organ circuit that I had built using a schematic from *Make Magazine* and a lot of discrete components from Jameco Electronics. I decided one day to see if I could fit/cram that board into the new found space inside of Beer2's can. Well, it wasn't easy to cut the board down to size, but with some Dremel work on the circuit board, rewiring of the circuits,



Beer2D2 meets Dalek.



Beer2D2 iPod.



Beer2D2 with his blinky lights.

and some new cuts on my hands, I got it in there. Plus, a Y connector was also added so the audio signals would go to both the audio amplifier and the LED color organ circuit.

I wrapped the whole board in a plastic sandwich bag and some electrical tape, so it wouldn't short out inside of the metal can. I also modified the original circuit by adding an output driver stage using TIP 42 PNP transistors. This way, I could drive a lot more LEDs. Beer2D2 got a bit more expensive at this point.

I bought some NTE Electronics' red, green, and blue waterproof self-adhesive LED strips. At the time, they were \$17 each. Plus, I bought some super bright LED modules at Fry's Electronics for another \$50, and yet more NiMh batteries to make another 12 VDC pack to power them.

But wait! That's not all! After the audio controlled blinky light upgrade, I decided to make him selectively autonomous. So, I also added a 7805 5 VDC voltage regulator for sensor power; a 7809 9 VDC voltage regulator for Arduino power; three HC-SR04 ultrasonic PING))) distance sensors — one for each drivetrain housing; and an Arduino microcontroller to read the PING)))

sensors and send the PWM control signals to the speed controllers. I actually wrote the Arduino code while his batteries were charging at RoboGames 2013.

A Very Simple Program

Basically, his simple program is always motivated to move him forward. However, if he gets within three inches of anything that the HC-SR04 PING))) sensors can detect, he will turn right if the left sensor is triggered, or he'll turn left if the right sensor is triggered. Beer2 will back up if the center PING))) sensor is triggered — all for one second in the desired direction to correct or clear his path.

I also added a 4PDT relay that is controlled with a toggle switch to manually select either the radio or microcontroller. To

switch which signal is used, the radio PWM signals are connected to the normally closed side of the relay, and the Arduino PWM signals are connected to the normally open side of the relay. The common pins connect to the speed controller's PWM input pins.

If the relay is off, the robot is in R/C mode; if the relay is turned on, the robot is in Arduino/PING))) sensor controlled autonomous mode. This is and has been Beer2D2's configuration since 2013. I'm sure he would

appreciate it if I upgraded his software one of these days.

Our Many Wonderful Adventures Together on the Planet Earth

Beer2D2 and I have been to all of the Bay Area Maker Faires and RoboGames events since 2010. We have also been to countless art, science, robot clubs, maker spaces, and frequent camping and road trips together. This little guy and I have personally met and entertained literally thousands of folks. I've seen him make a lot of people smile. Watching the looks on little kid's faces and having



More Beer2D2 fans.



Beer2D2 sing and dance with them has been beyond heartwarming for me.

At one of the trips to RoboGames, Beer2D2 and several of the robots from the Home Brewed Robotics Club did a parade. There must have been 20 children marching along with the various robots. That was a magical impromptu moment for all of us.

Another one of my other favorite things to do is have Beer2D2 drive around the booths at Maker Faires on Sunday night and see if we can get the incredibly tired folks that have manned those booths to smile after their epic ordeal of talking and demoing their own passions to thousands of humans. Beer2D2 almost always has.

One should always give back to the makers, is Beer2D2's way of thinking since they often inspire and teach us so much. When in doubt, always thank the makers!

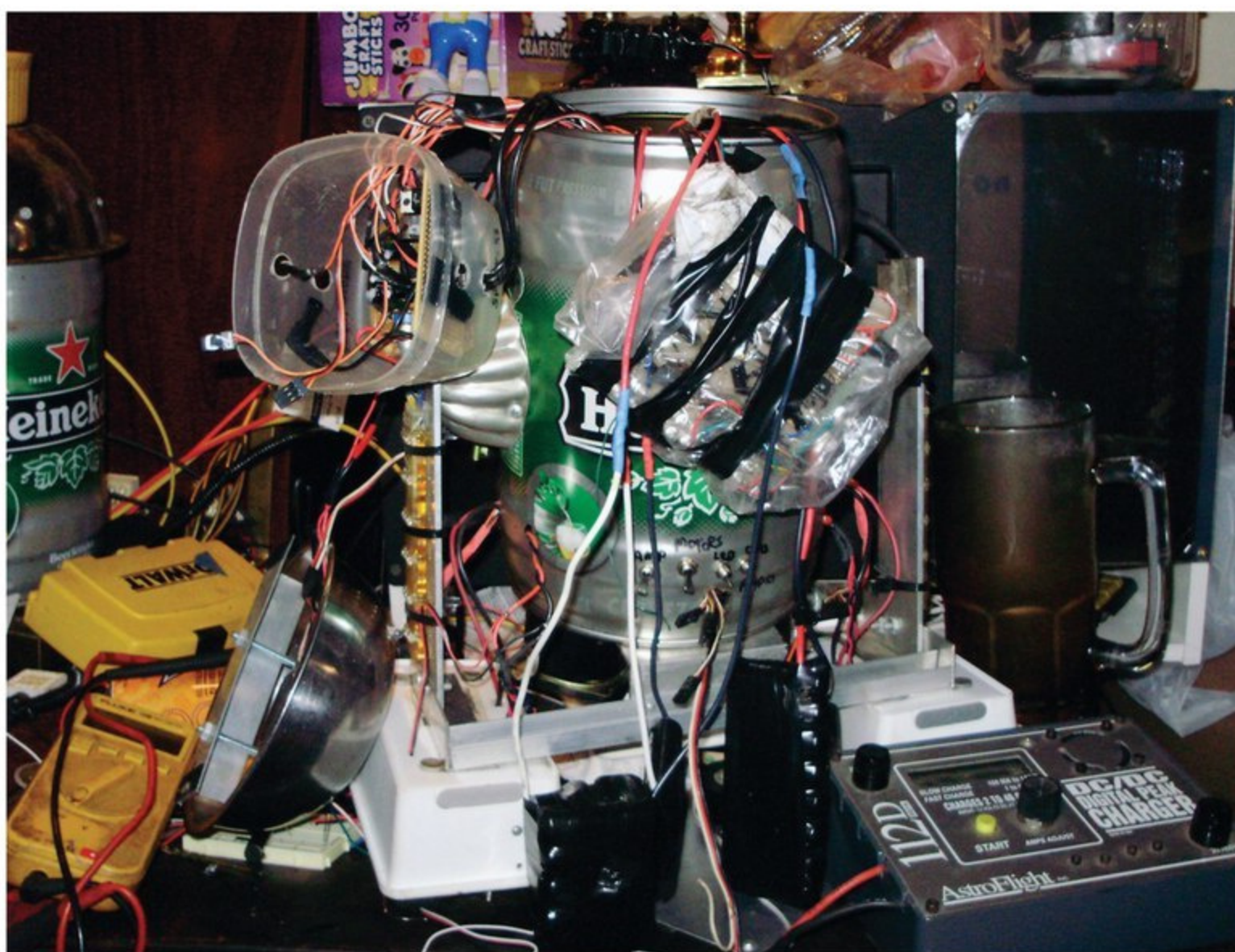
To our surprise, Beer2D2 was actually awarded the Editor's Choice award at the 2013 Faire. He got a blue ribbon. That's not bad for a previous random pile of Obtainium and a bit of science, robotics, new/used electronics, and stuff.

The Poor Little Guy Eventually Died. But He was Fixable.

In 2014, Beer2D2 really died and was completely unusable. After a bit of difficult surgery, wire cutting, soldering, and adding wire extensions, I was able to rebuild, buy new parts, and replace/build his current six failing battery packs that would no longer take a charge.

Beer2D2 performed well again at RoboGames 2015. With a bit of maintenance, he should be ready for his fifth trip to Maker Faire 2015. Of course, I still cut my hands reaching into his tin can when fixing him ...

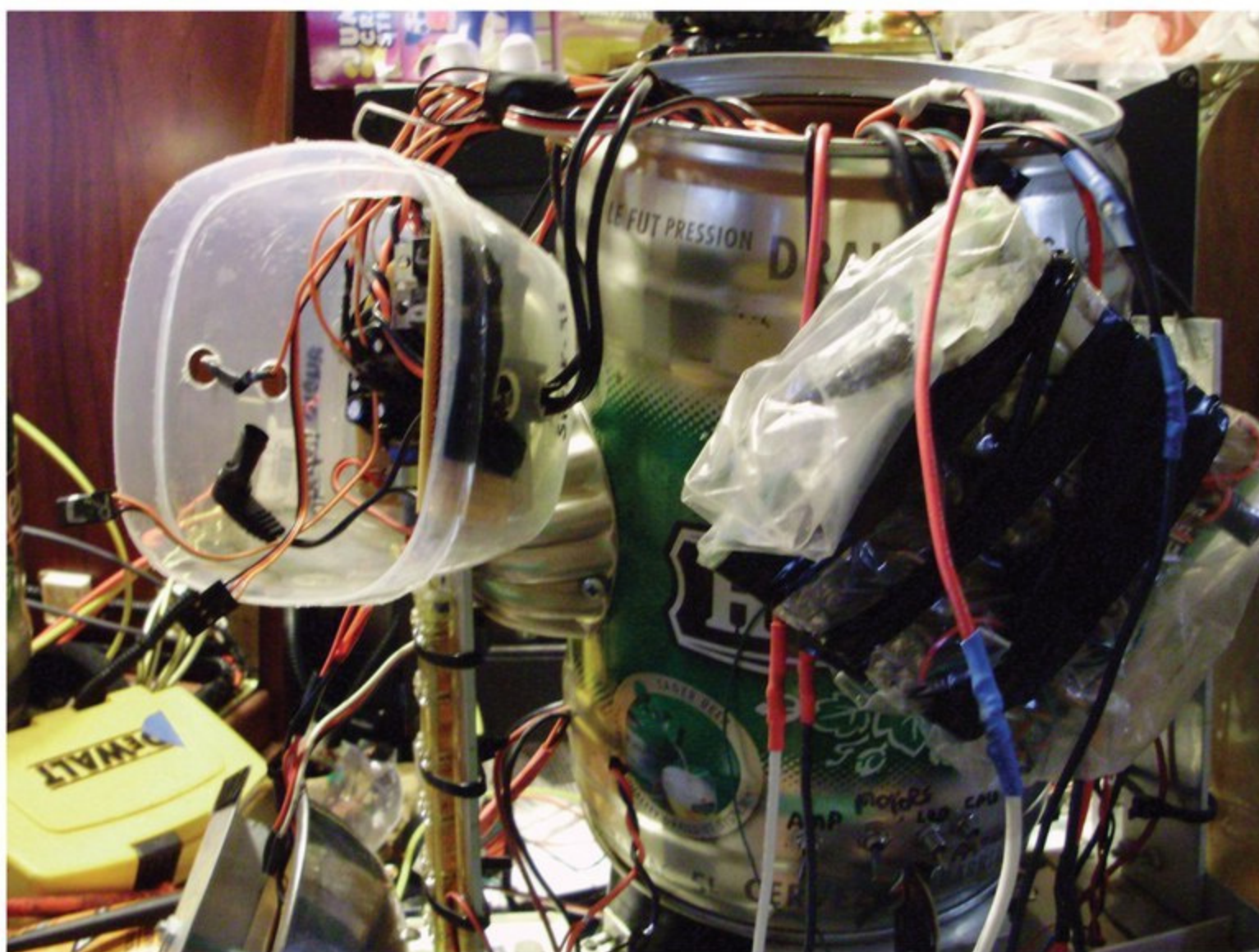
My human friend, Camp Peavy



Beer2D2 surgery.

is right. You can make a robotic friend out of anything.

Invent, Re-purpose, Recycle. It's good for you and it's good for the kids, too. **SV**



More Beer2D2 surgery.

CNC Part Creation Workflow

Part Creation



Load Mach 3

I use Mach 3 software to control all of my CNC machines. It's priced reasonably enough that even the hobbyist has professional quality software within his/her grasp.

The first step in the part creation process is to load Mach 3 (**Figure 2**) and hit the reset button. This should engage the drive motors and put you in the position to start the reference process.

Set Y Axis Reference

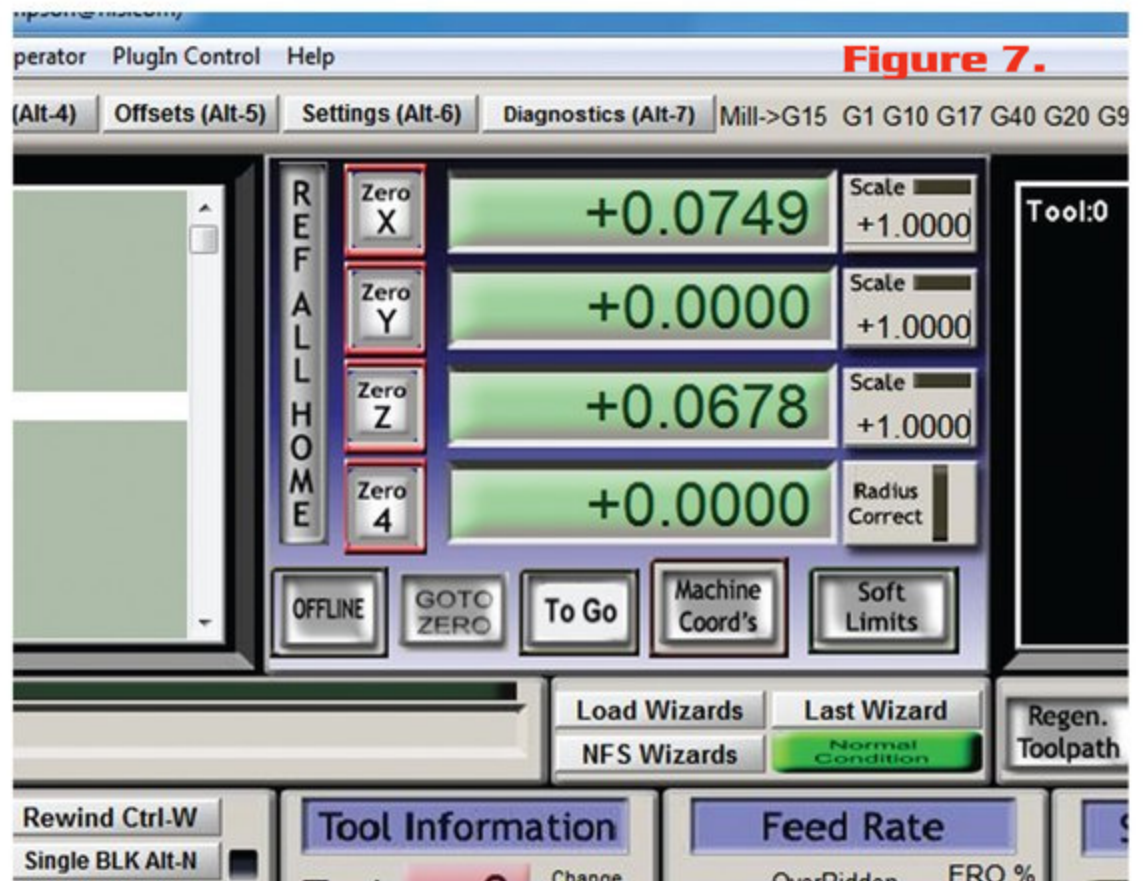
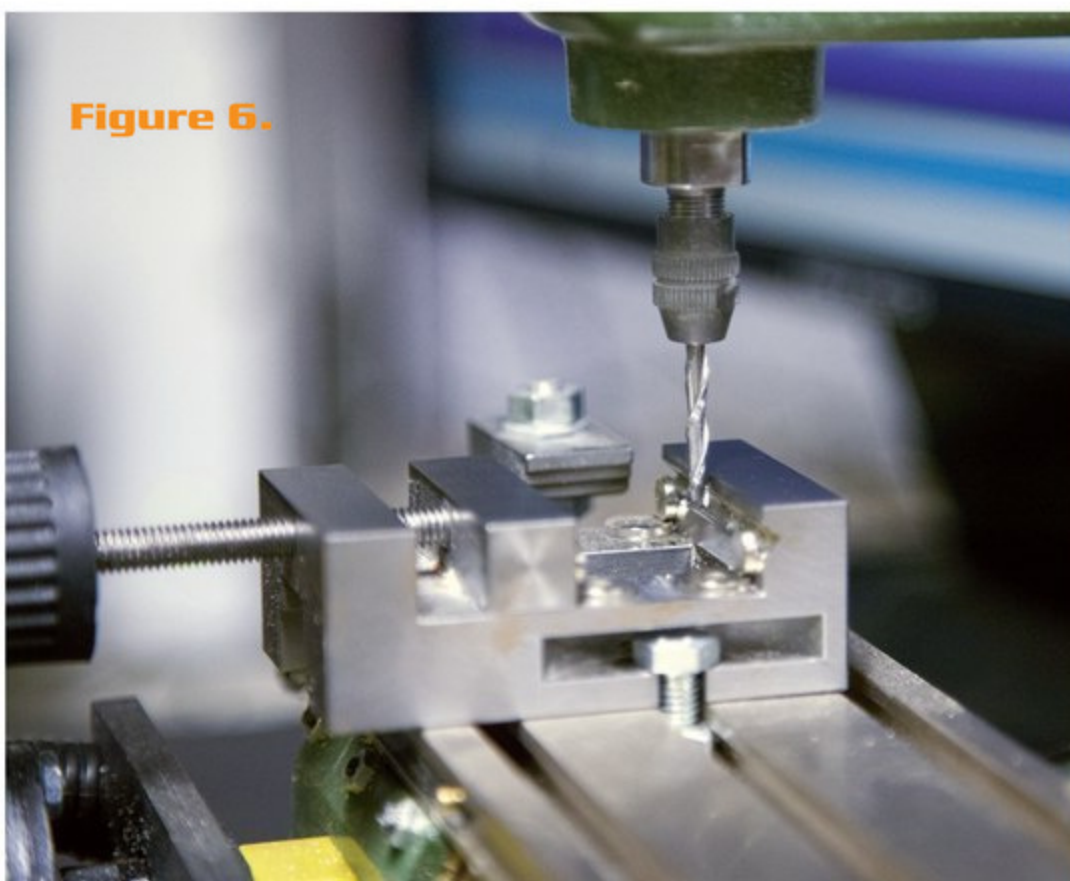
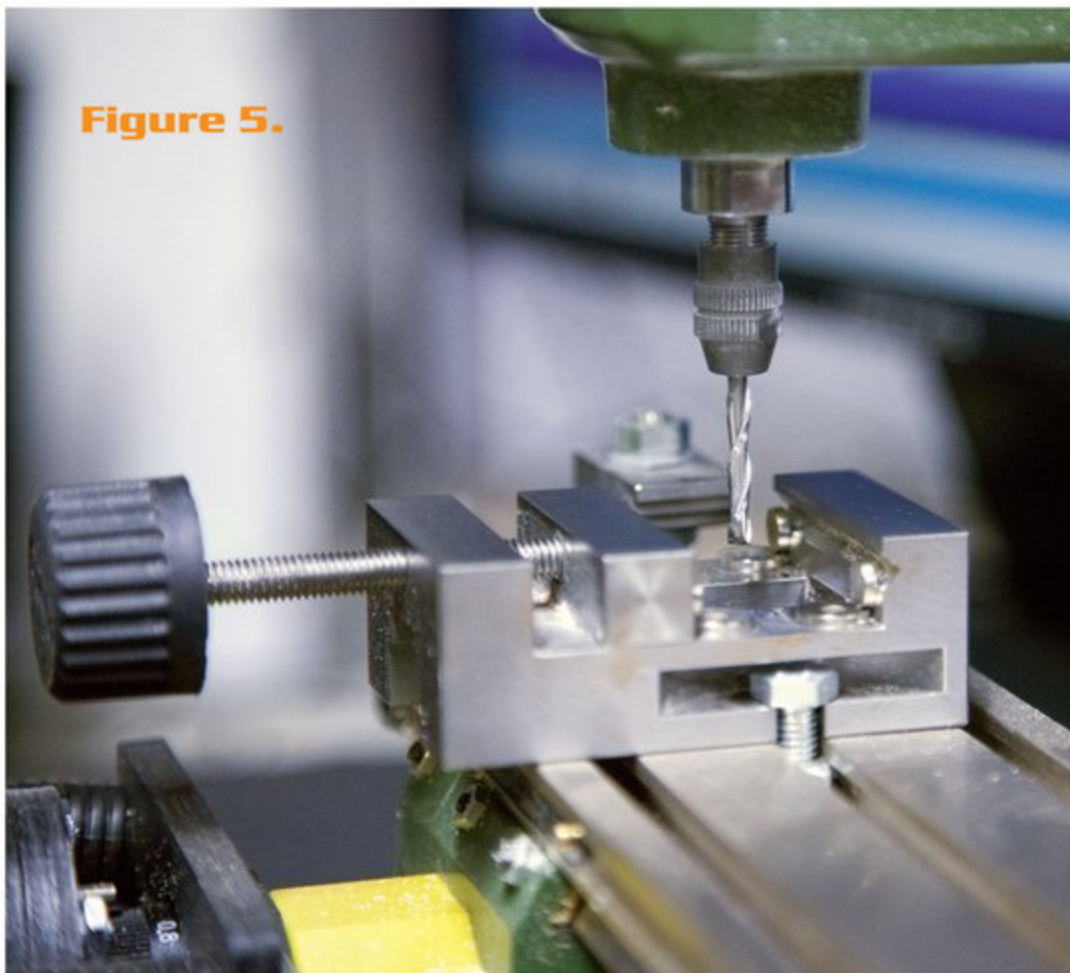
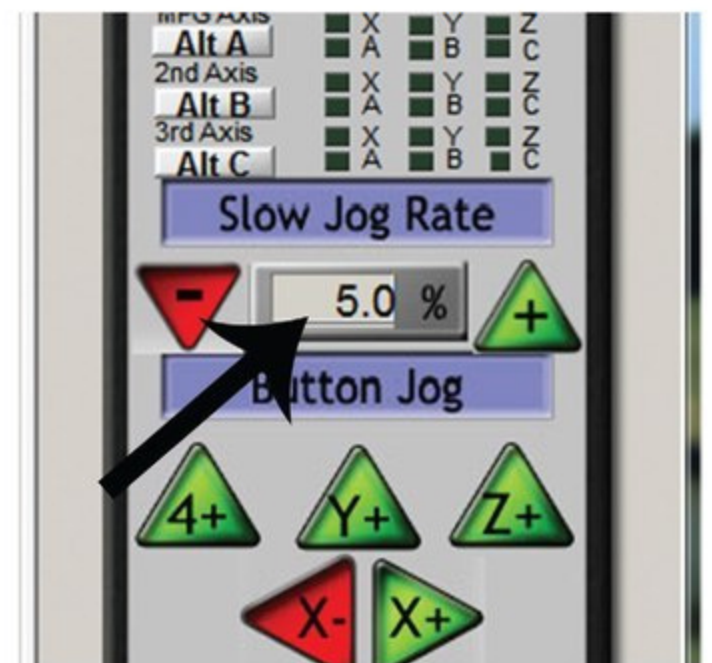
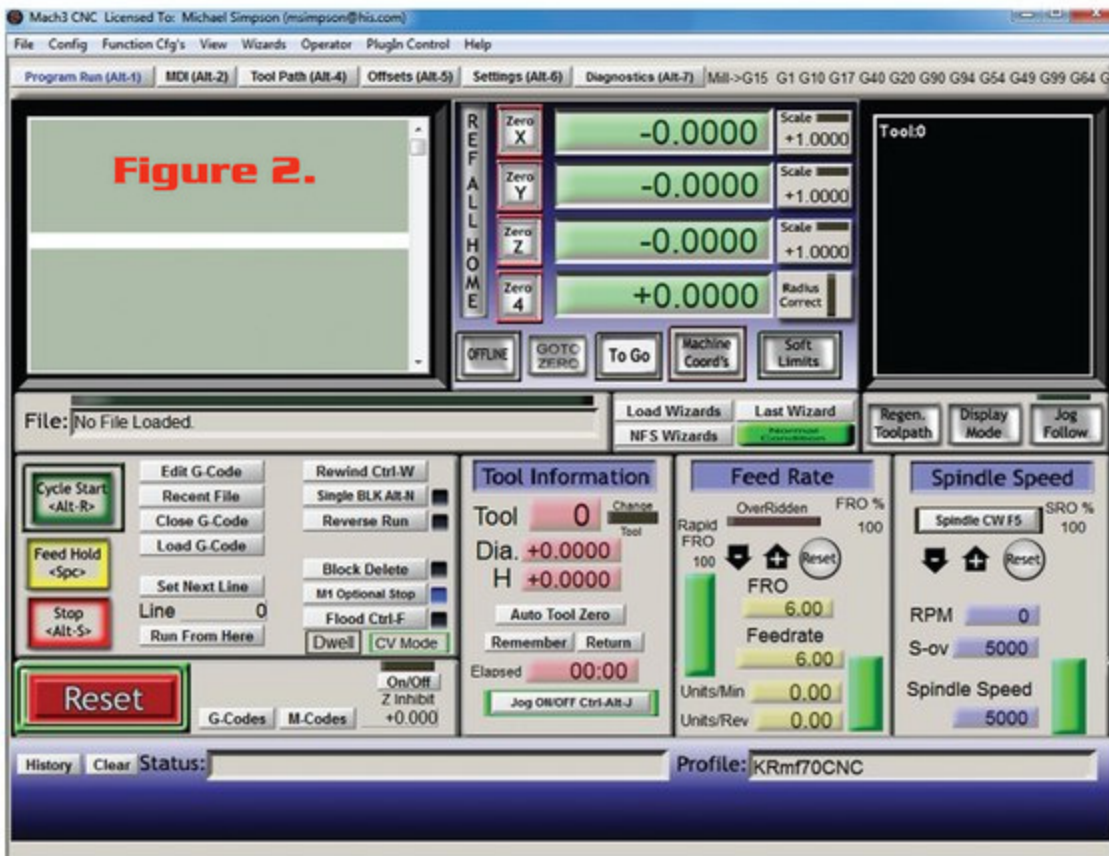
Start by loading the Mach 3 jog control shown in **Figure 3**. The jog control is toggled on and off by hitting the tab key. While you can use the jog control to move the CNC, we will be using the keyboard instead. We

By Michael Simpson

Post comments on this section and find any associated files and/or downloads at www.servomagazine.com/index.php/magazine/article/august2015_Simpson.

In this article, I will show you how I use the G-code files we created earlier to make the actual domino parts.

Before creating any part, you need to reference your X and Y axis to the stock you will be machining. For the domino box, you must reference the X and Y axis to the vise and vise stop. If you have not done so, you must set up and tune your vise before setting the X, Y, and Z axis references.



use the jog control to reduce the jog speed 5% of the CNC machine's maximum travel speed. This is done by entering 5% in the "Slow Jog Rate" field shown in **Figure 4**.

This will slow the machine down enough to allow us to fine-tune the reference points.

The next step is to load the milling bit. In this case, we will be using a 1/8" two-flute carbide router bit. Using the arrow keys, move the CNC so that the router bit is centered in the vise as shown in **Figure 5**. Use the page up and down keys to lower the bit so that it is just below the top of the vise.

Now, use the arrow keys to slowly move the bit until it just touches the rear jaw on the vise as in **Figure 6**.

Zero the Y axis in Mach 3 by hitting the "Zero Y" button shown in **Figure 7**. Next, move the bit up above the

Figure 8.

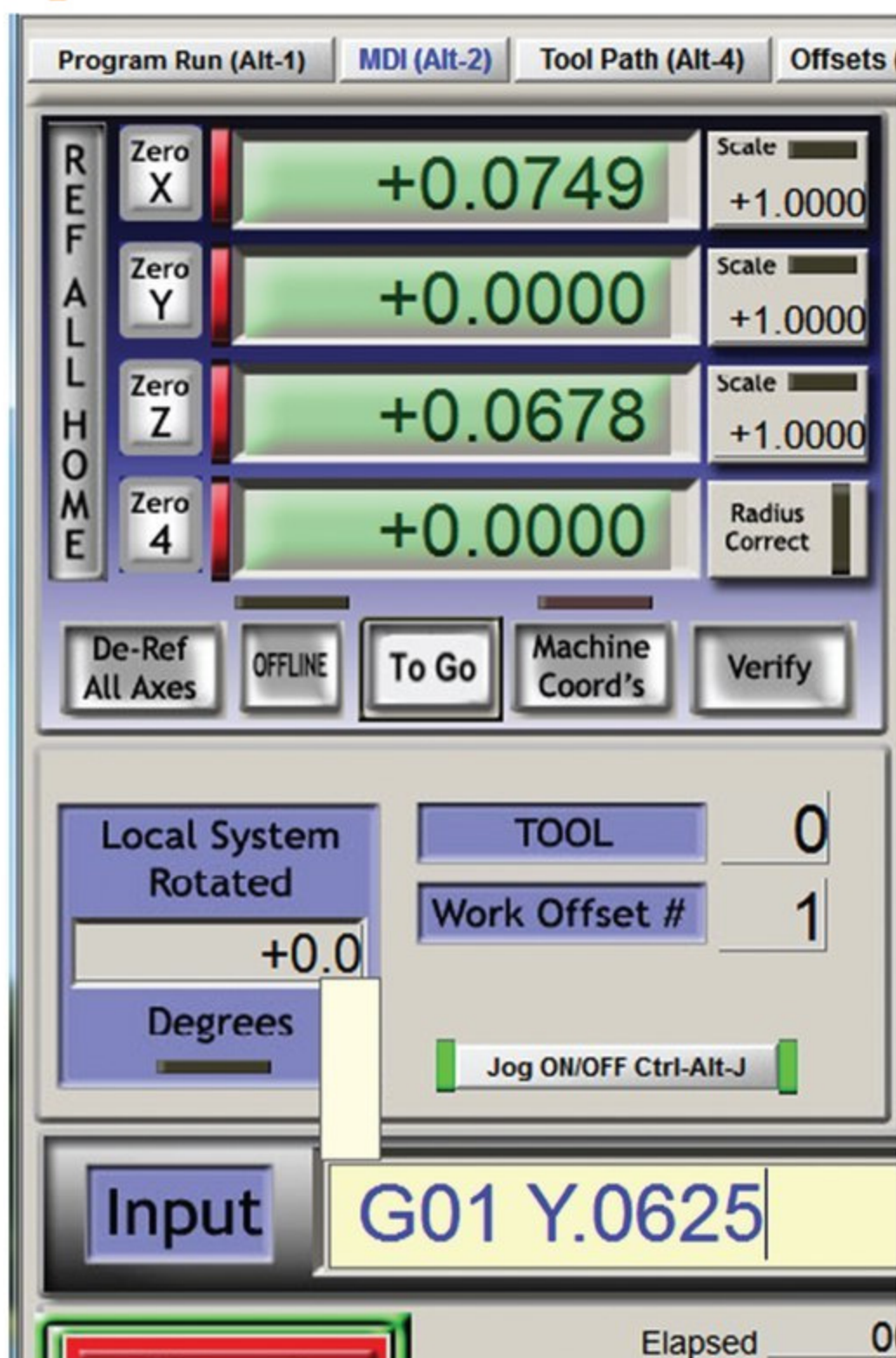


Figure 9.



Figure 11.

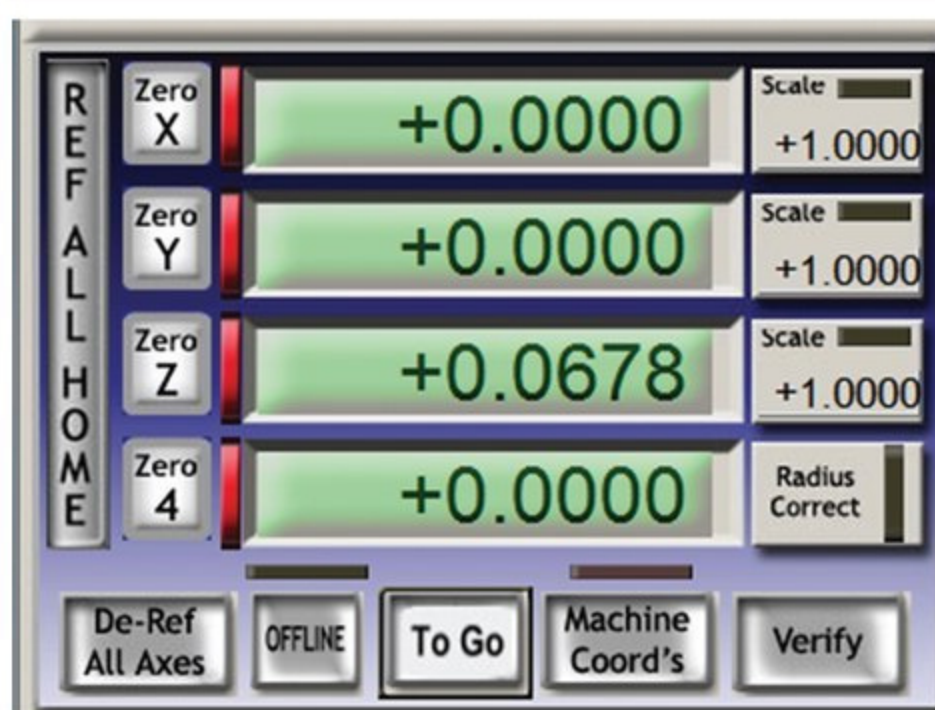


Figure 12.

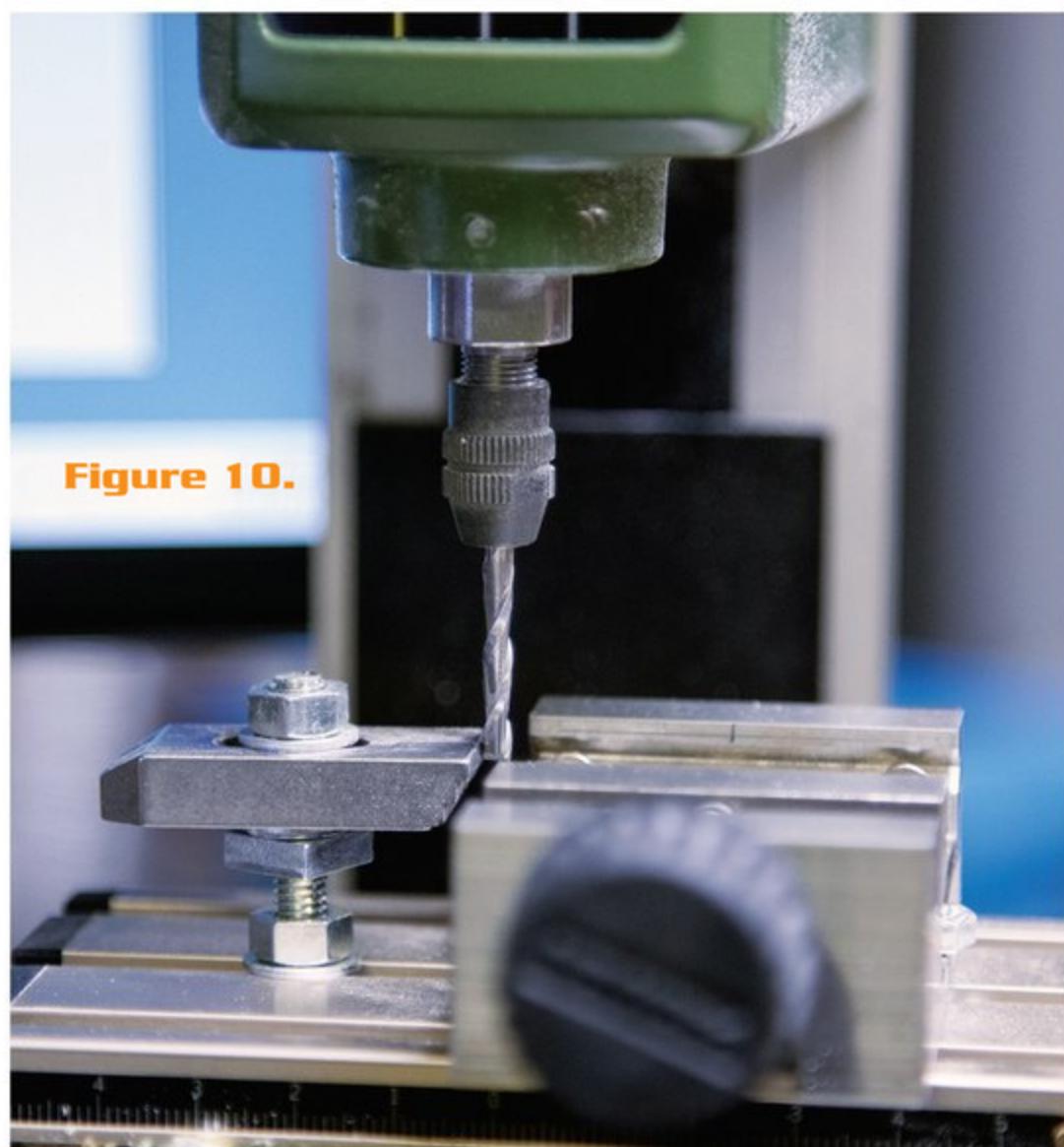
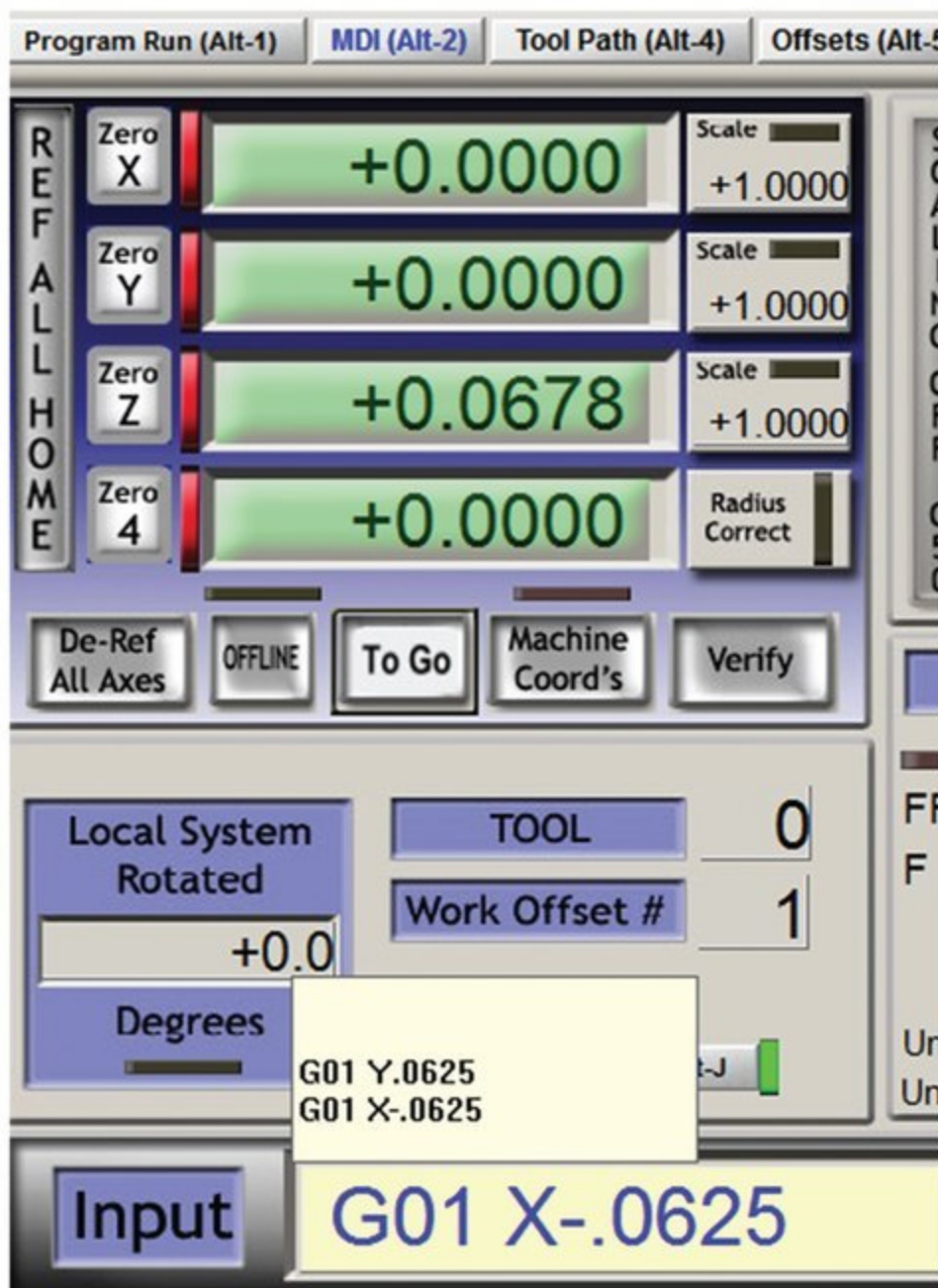


Figure 10.

vise so that it will clear it for the next step. From the MDI tab in Mach 3, type "G01 Y.0625" (**Figure 8**). Hit the enter key and the machine should move the bit to that position on the Y axis. We are doing this to center the 1/8" bit over the edge of the rear jaw.

To complete the process, zero the Y axis one more time by again hitting the "Zero Y" button shown in **Figure 9**.

The Y axis is now referenced.

Set X Axis Reference

The procedure for referencing the X axis is very similar to that of the Y axis. Start by moving the bit back to the center of the vise and below the top of the vise stop. Next, use the arrow keys to position the bit so it just touches the vise stop as in **Figure 10**. Hit the "Zero X" button (**Figure 11**). Next, raise the bit so that it will clear the stop, then from the MDI tab type "G01 X-.0625" as shown in **Figure 12**. Hit enter and the machine should center the bit over the edge of the stop. Zero the X axis one more time; the X axis is now referenced.

Figure 16.



Figure 17.

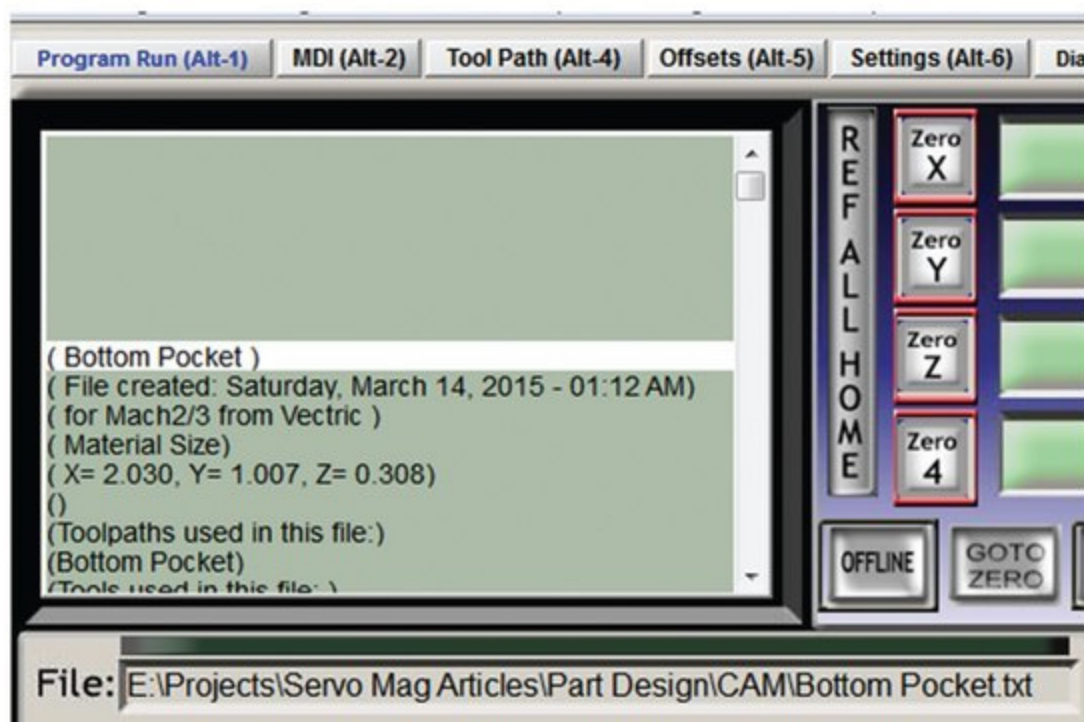


Figure 13.

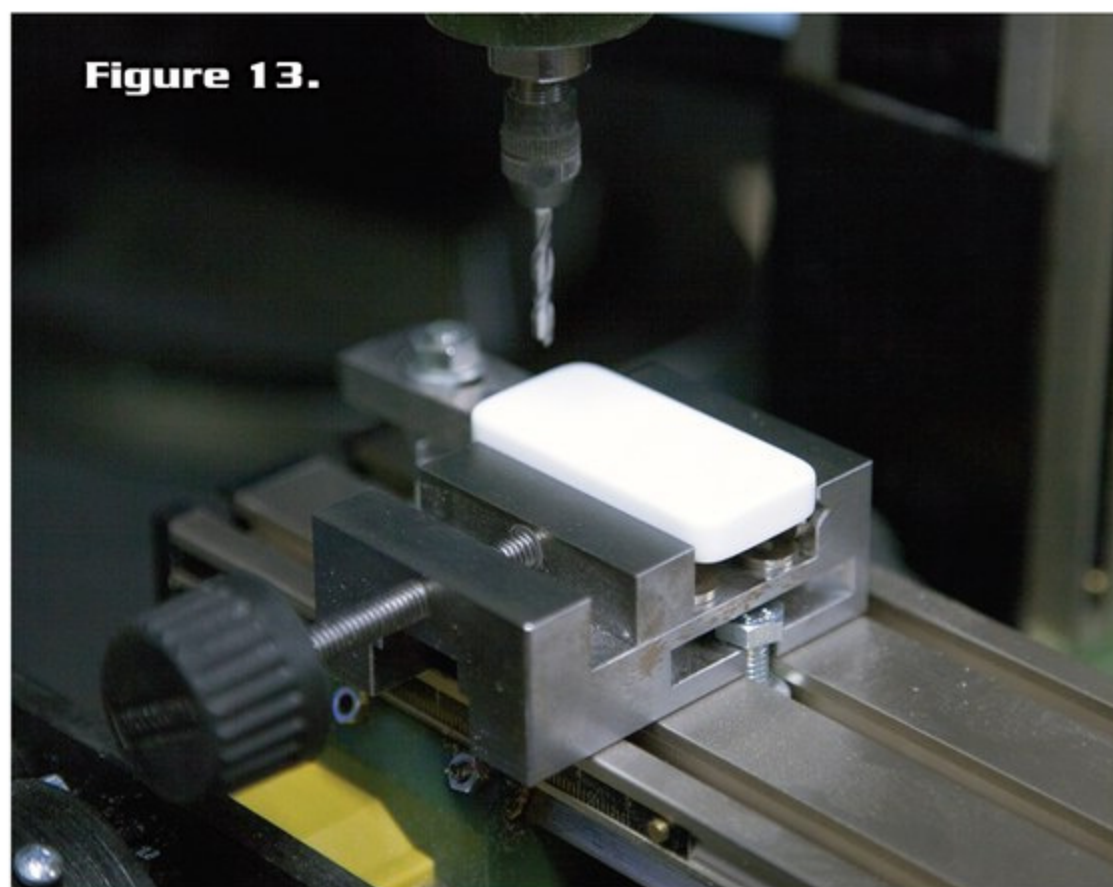


Figure 14.

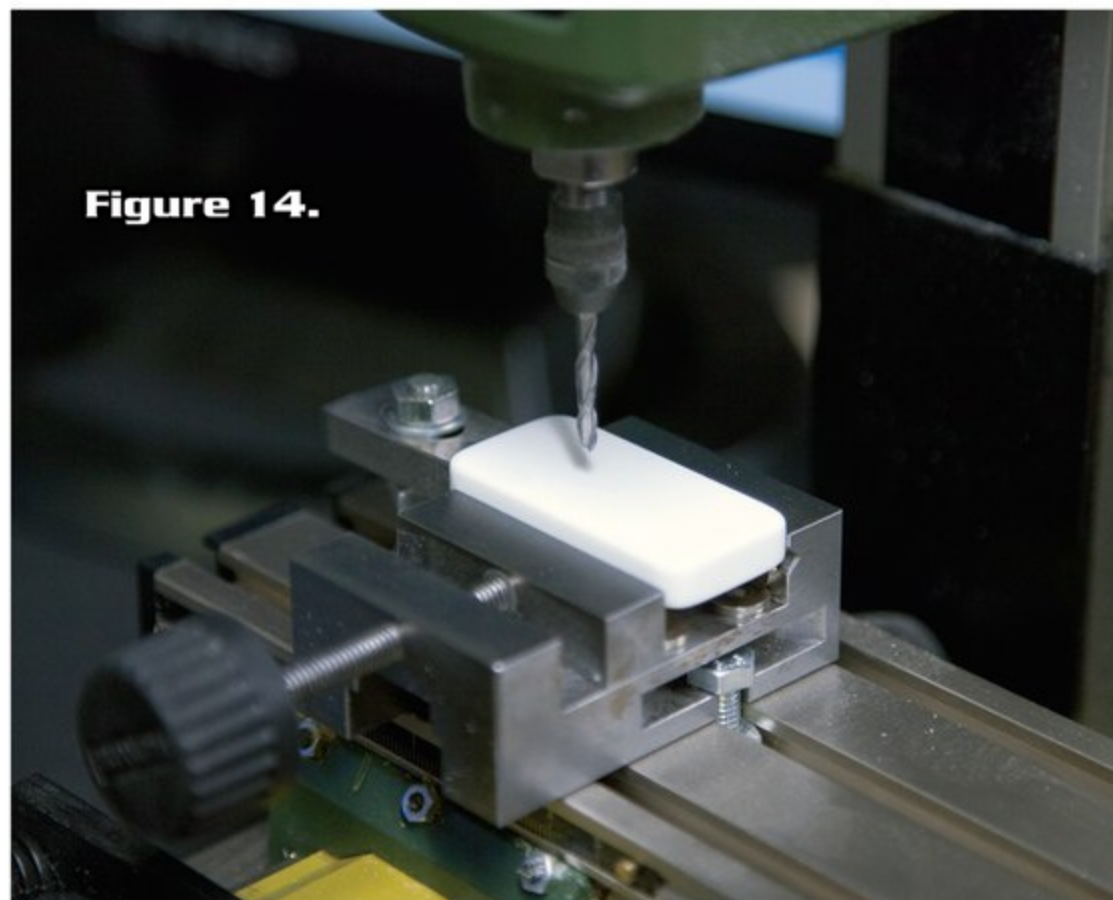
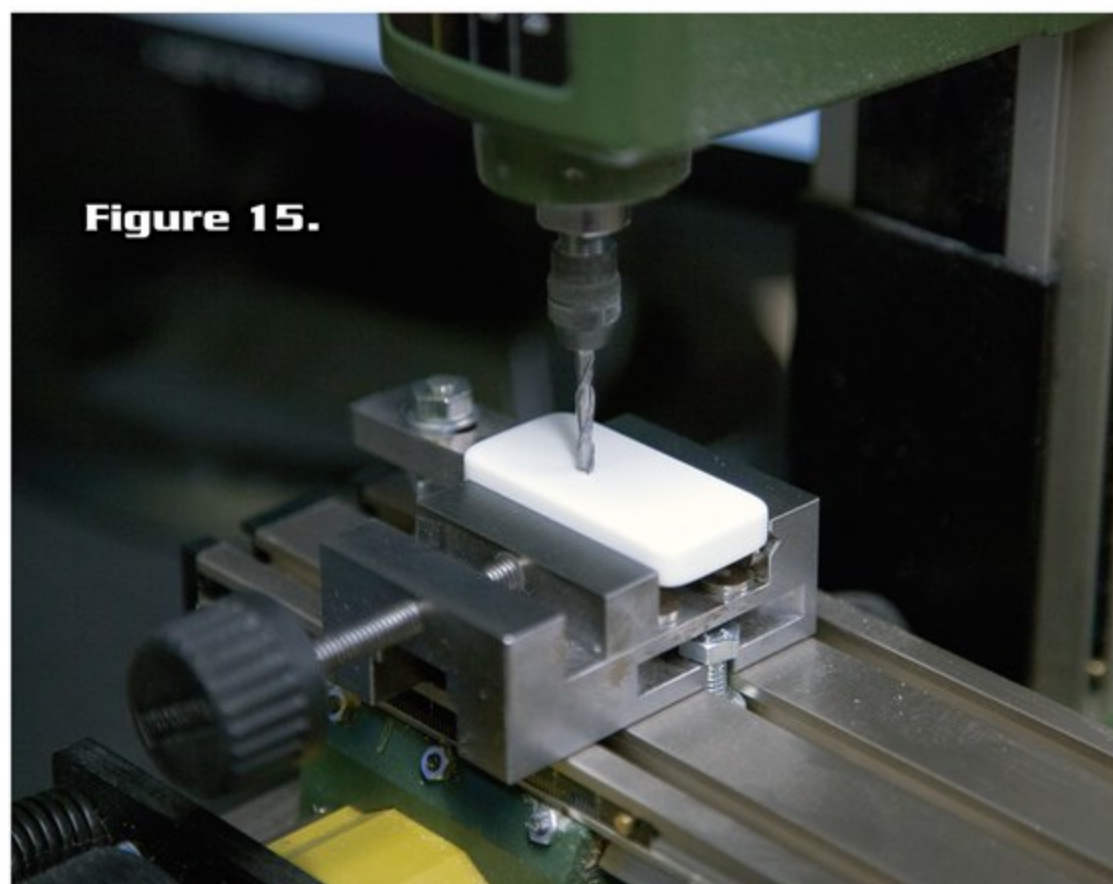


Figure 15.



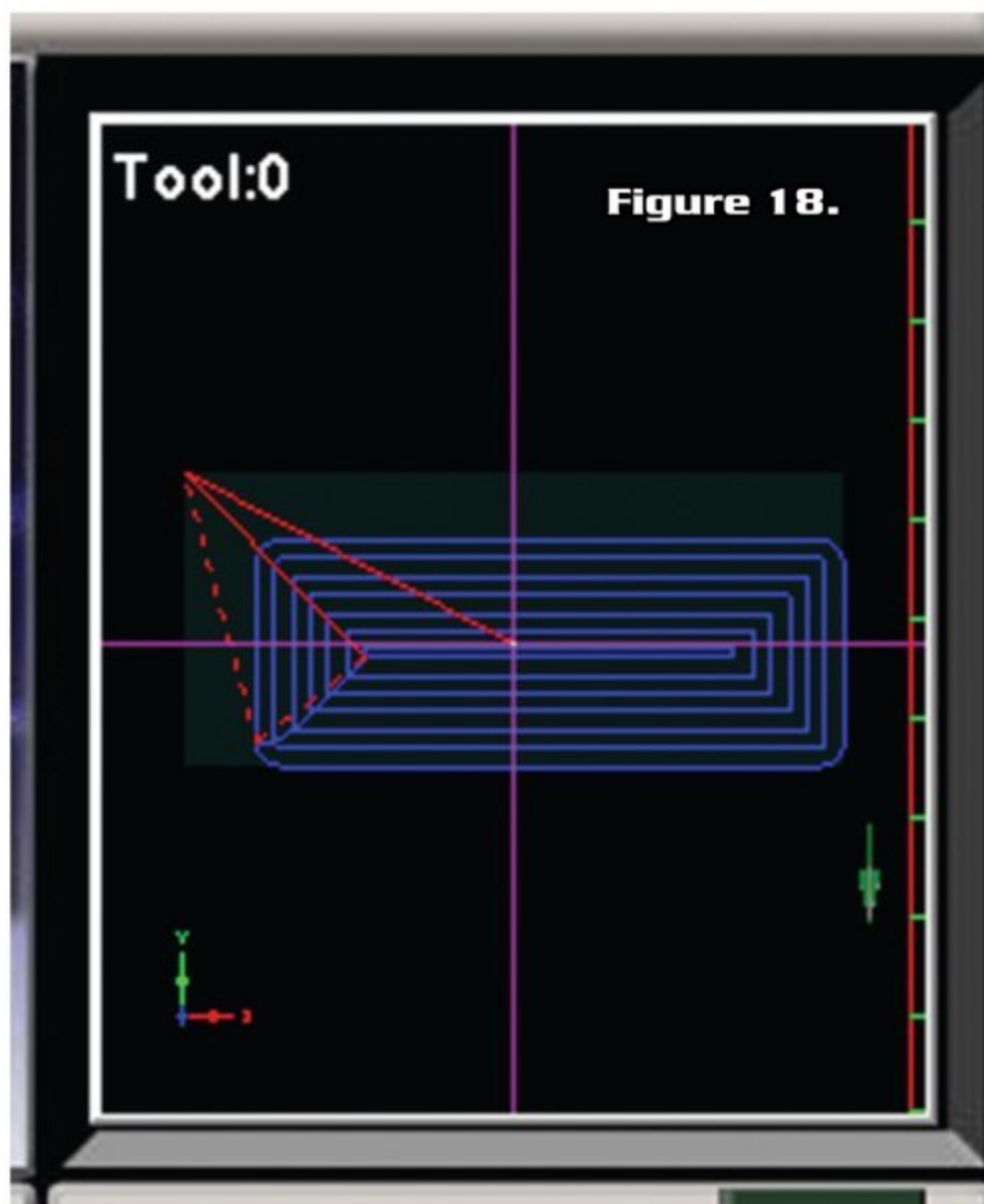


Figure 19.

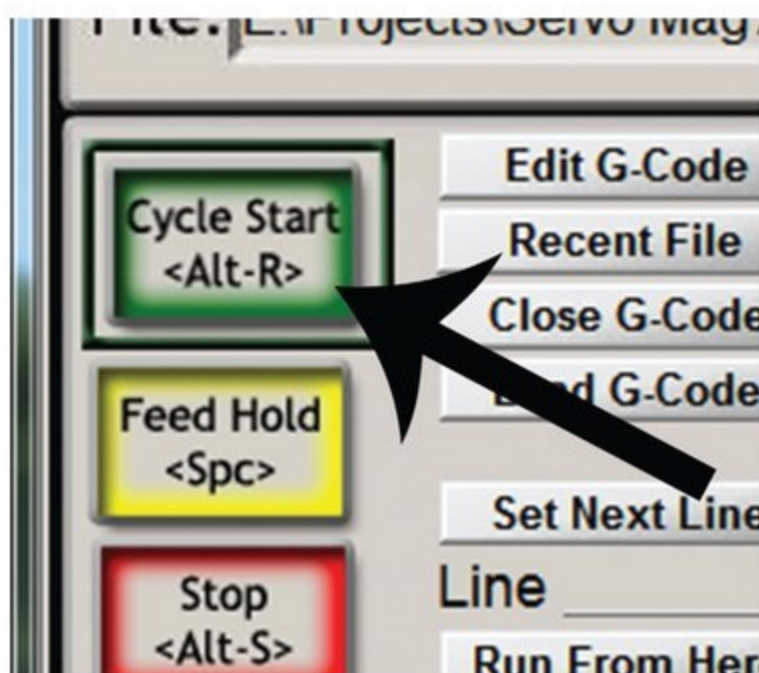
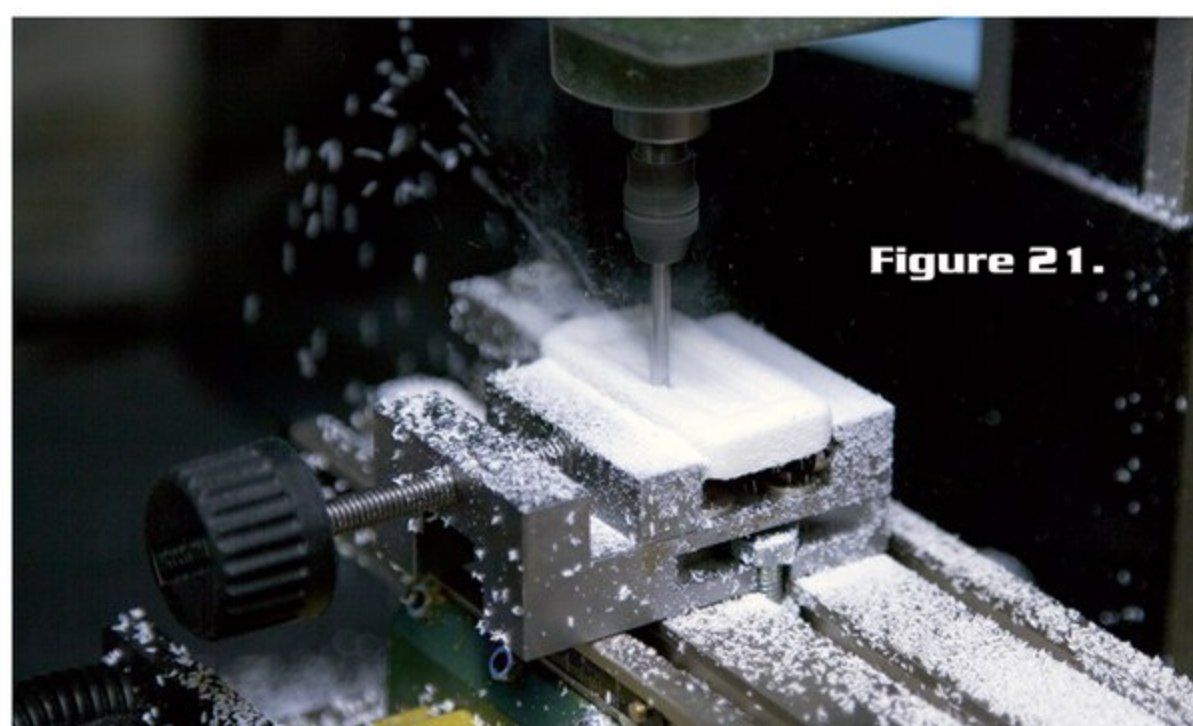


Figure 20.

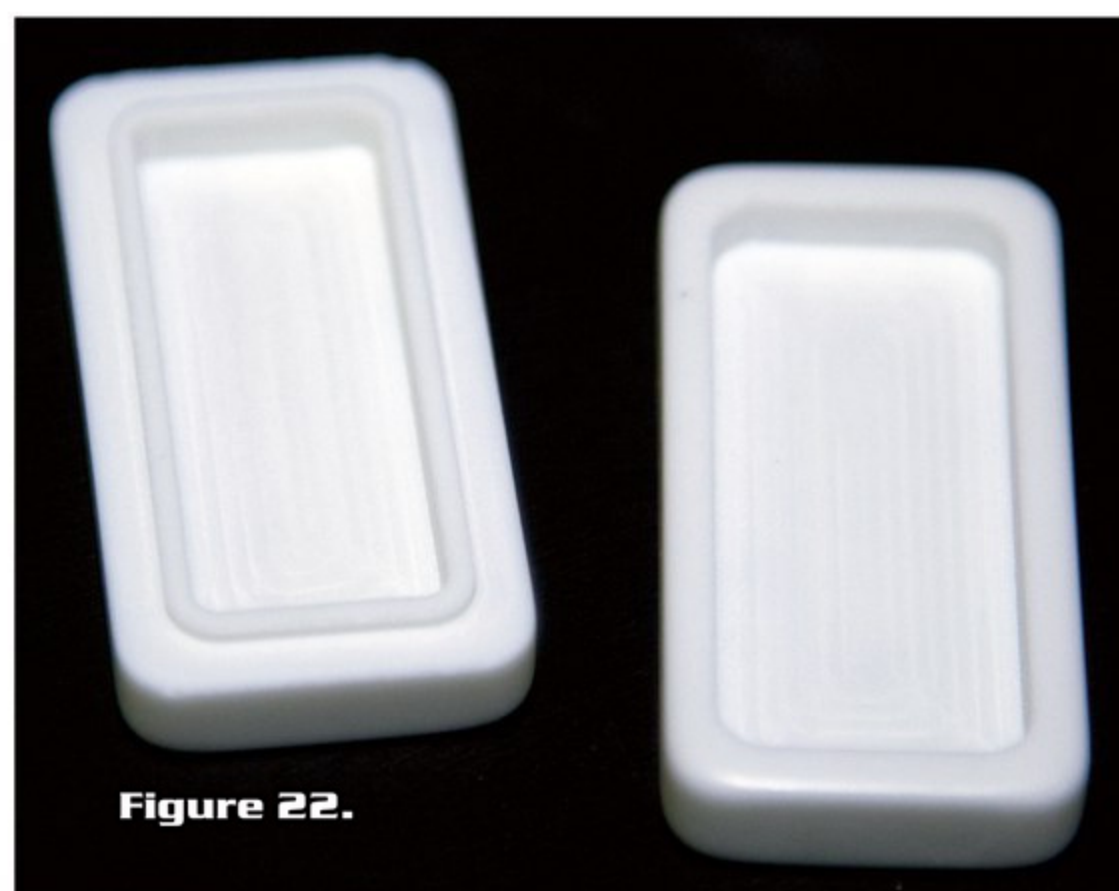


Notice that the X axis was moved a negative amount. This was done because we were moving the machine to the left of its coordinate system. If we were referencing the other side of the vise, we would have used a positive value.

Set Z Axis Reference

In order to zero the Z axis, you must load the domino stock into the vise (**Figure 13**). Make sure it is up against the vise stop. Move the bit so that it is roughly in the center of the stock (**Figure 14**). Using the page up and page down buttons, slowly move the bit so that it is just touching the stock as shown in **Figure 15**.

Hit the "Zero Z" button (**Figure 16**). The Z axis is now referenced and you are ready to mill the stock.



Milling the Domino Bottom

Last month, I showed you how I created the G-code for the top and bottom of the domino box. With a domino loaded face down and tightened into the vice, it is time to load the G-code for the bottom. The bottom G-code file was saved as "Bottom Pocket.txt." That's the file we will load into Mach 3 now.

Once the file is loaded, you will notice two major changes to the Mach 3 interface. The first (**Figure 17**) is a list of the G-code commands that the program is going to run. The second (**Figure 18**) is a rendering of the tool path.

In order to start the process, we need to do the following:

- Make sure the bit is raised off the stock and near the center.
- Make sure the center of the cross hairs on the tool path rendering is pointing to the same point on the stock where the bit is positioned over the stock.
- Set the speed on the MF70 to 18 (**Figure 19**).
- Turn on the spindle motor (**Figure 19**).

The final step is to start the job. This is done by hitting the "Cycle Start" button shown in **Figure 20**. It is important to note here that some machines will start the spindle automatically. Some will even set the speed of the spindle for you. Once the job has started, the bit will start cutting the stock as shown in **Figure 21**. As the job progresses, you will see Mach 3 cycle through the commands in the G-code screen. When the job is complete, the bit will move up away from the stock.

When this happens, turn off the

spindle. On some machines, the spindle will stop automatically.

Milling the Domino Top

The domino top is done exactly like the bottom. Since your machine is already referenced, all you have to do is load a fresh piece of domino stock and load the G-code file for the top. Repeat the same processes and hit the Cycle Start button to begin the job.

Once complete, stop the spindle and remove the stock.

Conclusion

The top and bottom shown in **Figure 22** fit together so well that I have to use a screwdriver to pry them apart. At this point, I would go back to the drafting stage and increase the pocket size to remove just a little from the outside of the lip to loosen the fit. This can also be done directly in the CAM software.

The machine referencing may seem a little tedious but, in reality, it goes very fast. In addition, once referenced, you can cut several parts without making any more adjustments.

Referencing the Z axis can also be automated by adding a depth probe to the CNC. I am currently working on a upgrade book for the KRmf70 CNC where I add spindle control and a depth probe.

Final Thoughts

I hope this series has given you a better understanding of part design and creation on a CNC. Keep in mind that this process is different from full 3D design and milling. 3D milling is done with a V-bit (or bull nose bit) and in a series of overlapping passes on the stock. **SV**

The Easy Way to Design CNC Front Panels & Enclosures

You design it
to your specifications using
our FREE CAD software,
Front Panel Designer

Free
Front Panel
Designer



We machine it
and ship to you a
professionally finished product,
no minimum quantity required

- Cost effective prototypes and production runs with no setup charges
- Powder-coated and anodized finishes in various colors
- Select from aluminum, acrylic or provide your own material
- Standard lead time in 5 days or express manufacturing in 3 or 1 days

**FRONT PANEL
EXPRESS**

FrontPanelExpress.com

If you want more information on my KRMx02, KRmc01, or KRmf70 CNC books, you can find them at www.kronosrobotics.com. You also can download an early version of the KRMx01 CNC router book for free at www.kronosrobotics.com/krmx01/index.shtml. If you have any MF70 questions on this or previous articles, please use the forum at <http://forum.servomagazine.com/viewtopic.php?f=49&t=17107>.

The **DARPA** Robotics Challenge Finals 2015

On Saturday, June 6th of this year, I had the joy of attending what the *Huffington Post* called the "Super Bowl for Robots" — the DARPA Robotics Challenge Finals (DRC). You see, I am a long time roboticist, having built my first robot over 40 years ago and not stopping since. When I heard a few months back that the DRC finals were being held in Pomona, CA — about 40 miles from my home — I let out a child-like scream of delight! I literally bounced up and down with joy. I could not contain myself! The DARPA Robotics Challenge Finals were being held less than an hour drive from my home! Not only would I be able to watch this riveting (excuse the pun) event with my own eyes, but I would be able to talk to the brilliant robotics engineers that designed, built, and programmed these steely mechanical men.

But there was MORE! When I arrived — anticipating to be wowed and awestruck by the competition — I found that there was more than a robot "Super Bowl" planned. I found myself in the middle of an ongoing robotics "half-time" in the form of an out-of-this-world robotics exposition.

This robot expo was bigger than any I had ever attended. It was chockfull of not just the companies and schools that were competing, but robotics vendors each exhibiting, promoting, and demonstrating robots of every size, shape, and configuration. I suddenly found myself in robot heaven!

I rolled into the Fairplex at about 9:30 in the morning to see the Robotics Challenge. I came to the DRC for several reasons: the first one being to enjoy watching 24 robot building teams compete for a grand prize of \$2 million; and the second one, to promote my own college of robotics.

If you have been living under a rock, you might be asking "What is DARPA?" When I first heard of DARPA, I assumed because of the ".mil" of the website it was the part of the US military installation building and researching Terminator-esque robots. I had guessed that their sole purpose was to militarize robots so that we could replace

soldiers with T600s and human-guided military drones with autonomous hunter-killer robots. Even DARPA's mission statement, "Creating breakthrough technologies for national security" might lead one to believe that "Skynet" was in the works.

Robot Heaven: The Technology Expo at the DRC

Going to the DRC, one entered the event grounds walking into one of the biggest and most advanced robotics and technology exposition in perhaps the world. With over 70 exhibits and demonstrations put on by robotics organizations, colleges, clubs, and first responders, it was like a robot Disneyland! Still, I was not there just to gawk at the latest cutting-edge robotics technologies, I was there actually on a three part quest. First, to support the Robotics Society of Southern California, to which I had belonged for more than 10 years. Second, to experience the DRC and its expo as a attendee, and third — and perhaps most importantly — to showcase the California Institute of Robots: a robotics engineering college that I am working to open in Southern California.

The "Super Bowl" of Robotics

Find additional content and post comments on this article at www.servomagazine.com/index.php/magazine/article/august2015_Messerschmidt.

By Thomas Messerschmidt
Photos courtesy of Walter Martinez



After paying my \$10 parking fee I rolled up to the first parking attendant explaining that I had a robot to exhibit at the expo. He waved me forward to the side parking lot. At that parking lot, I pulled into an empty space, then my entourage of my wife, daughter, and father-in-law gathered my robot, my flyers, and then proceeded in.

I quickly made my way to Robotis' booth which was more like a circus tent with table after table of exhibits. Robotis didn't just have a booth. They had a SET of booths and a HUGE central tent showing off every product, every robot, and every technology they owned. There were 10 cute foot-tall bipedal robots on a center table, each robot on its own LED-lit stage, all dancing in perfect sync to Psy's Gangnam Style.

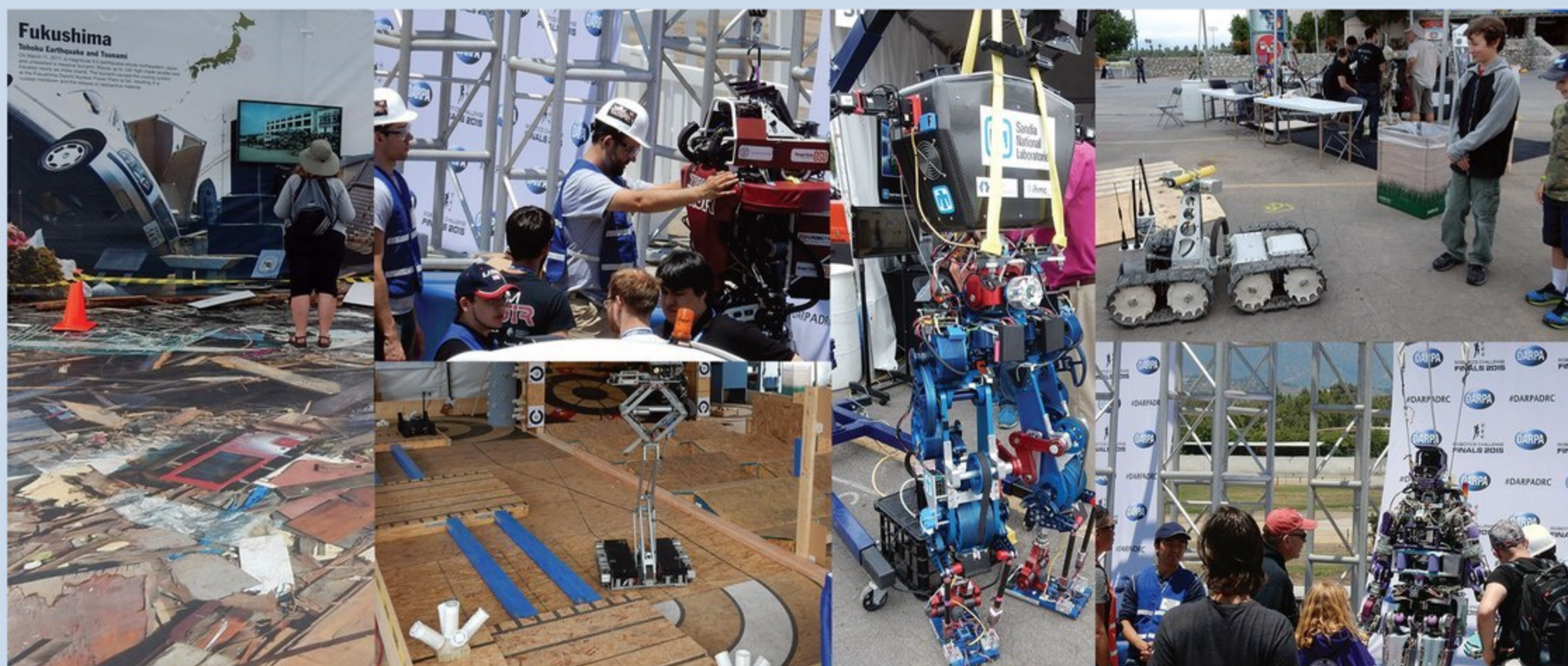
There were groups of people from my own club helping children assemble robots that looked like huge mechanical beetles. There were people conducting "Sumobot" contests. There was a replica of Robotis' own DRC competition robot, THORMANG 2 (15th place with three points) that was strapped to a chair to allow attendees to snap photos of their loved ones sitting with this cog-driven DRC competitor.

There was a giant claw balancing a huge metal ball.

There were servo motors of every kind being shown off. There were even companies that used Robotis' special servomotors boasting their creations, including a robotic device that could feed those who could not feed themselves. I am assuming it would be for handicapped adults, although I imagined that it might have been very helpful for feeding my own child when she was an infant.

To the left of the Robotis tent was SRI International's display, complete with two bigger-than-life steel and plastic humanoid automatons plodding along relentlessly on treadmills, not missing a step. I watched these stately sentries for a few minutes, then walked back to again be mesmerized by the many technologies on display at Robotis' exhibit.

After spending way too much time soaking up the awesomeness of their multitude of robots, I said hello to my fellow club members of both the Robotics Society of Southern California and the Riverside Robotics Society. I then set up my table for the California Institute of Robots and stood up my life-size robot college mascot, Robot Betty9. Next, I put flyers announcing the robotics college in my wife's and my father-in-law's hands so they could help promote the college. I then grabbed my nine year old



daughter, Maya by the hand and began exploration into robotics terra incognito — the DRC Expo.

At the expo, we wandered from booth to booth. I talked to engineer after robot building engineer. I talked to the engineers that built the DRC robots. I talked to NASA's engineers about their robots. I talked to a man with a life-size white robot parrot on his shoulder that squawked and tried to bite my finger. I saw a robot automobile driver that could theoretically drive any car and might just kill the Google car long before it arrives in showroom floors.

My daughter and I walked the long central expo corridor which highlighted DARPA, as well as many robots, engineers, and inventors over the century. I saw and chatted with friends and associates that I knew from the Riverside Robotics Expo that our local club puts on every year. I stopped with my daughter so that she could "be the robot" and perform a few of the robotics challenges herself with a manually controlled robot hand.

Some of the bigger names that were there at the expo included: MIT, Carnegie Mellon, Lockheed Martin, Cornell, SRI, National Geographic, the R2 Builder's Group (R2D2 replicas ARE big in my eyes), GE, iRobot, IEEE, and a gaggle of kids and mentors from the FIRST robotics organization.

NASA Created a Fembot and "Her" Name is Valkyrie

As I was making my way from booth to booth absorbing every nuance of leading-edge robotics technology, I found myself at NASA's exhibit. As I was watching an engineer demonstrate a disembodied robot hand with a silent beauty and fluid motion, I saw out of the corner of my eye a large humanoid robot standing just a few feet away from me. I started to question the engineer in the booth about this large-as-life human-looking machine.

I took a closer look at the six foot tall human-esque android. It kind of looked like an astronaut in a space suit — with a thong and an Iron Man-like light pulsing blue in its chest. I did a double take. "She" definitely looked like a

female astronaut wearing a space suit.

Now, I build robots as a hobby and part-time business, and the pride of my robotic creations is my fembot Robot Betty9 made from a plastic mannequin, a dozen overpriced servos, and an EZ Robot microcontroller. So, when I saw that NASA built a fembot, my jaw dropped to my chest. I stared at the lovely robotoid lady for a few seconds, and finally, in front of everyone, I burst out laughing! NASA had built a curvaceous feminine humanoid. As the NASA engineer explained, "You see, we ran out of room to put the ball screw motors, and so extended the chest cavity. We've gotten a bit of flak for that."

Strange Looking Robot Contenders

Throughout the day, I went back and forth from the expo and into the grandstands to witness the actual DRC competition. The competing robots came in many human-like shapes and sizes. From Tokyo, Team HRP2's robot was a super hero-like Transformer robot, while some robots — for example, Tartan Rescue — appeared more like mechanical gorillas. Team Aero's robot had four legs, and I remember at least one robot being able to roll around on its wheeled "shins" and then stand up to walk when needed.

Team Grit's robot, Cog-Burn — with its pencil-like appendages — resembled a walking stick insect. Some of the robots looked like gentle astronauts, while others resembled perilous killer robots from your worst nightmares. Various robots took a few slips and falls though. (More on that later.)

Introduction to DARPA and the DARPA Challenges

By taking a few minutes to read about DARPA from their website at darpa.mil — aside from the obvious DoD (Department of Defense) weaponry goals — I learned a few new things.

The birth of DARPA goes back to 1957 with the launch



of the USSR's Sputnik spacecraft and the beginning of the space race. It was in February 1958 that DARPA — then known as "ARPA" (the "d" was not added to the name until '72) — was formed as a reaction to that Russian milestone.

I also learned that DARPA seeks: 1. "An investment in developing cyber security systems to enhance and ensure that our national security is not compromised;" 2. A Biological Technology office that aims to "outpace the spread of infectious disease;" and 3. DARPA is "working to achieve new capabilities by applying deep mathematics; inventing new chemistries, processes, and materials; and harnessing quantum physics." Wow! Quantum physics!

DARPA's Five Robotics Programs

The **Autonomous Robotic Manipulation (ARM)** program has the goal of creating autonomous robotic arms with hands complete with nimble fingers that can operate with the dexterity of a human's.

The **Legged Squad Support System (LS3)** program has the goal of developing semi-autonomous four-legged pack-animal-type robots to be utilized on the battlefield by both Marines and soldiers. Boston Dynamics' Big Dog robot was one such bot.

The **Maximum Mobility and Manipulation (M3)** program is pretty much what its name implies. Its goal is to "... create and demonstrate significant scientific and engineering advances in robotics ..." A quadruped robot created by Boston Dynamics called the Cheetah was funded by this program. Clocked at running over 29 MPH, it is billed as the fastest legged robot in existence.

The **Phoenix** is a space science program. It was designed to create mission-critical satellites that could operate for their entire existence in outer space in a geosynchronous Earth orbit (GEO), without repairs or upgrades.

Then, finally we come to the fifth program: the **DARPA Robotics Challenge**. The DRC came about as a reaction to Japan's Fukushima nuclear power plant meltdown in 2011.

The competition tasks were chosen based on the problems that occurred during the attempted shutdown of the plant and containment of the radiation.

What Exactly is the DARPA Robotics Challenge?

The DRC is a competition designed around the idea that when a natural or man-made disaster strikes — such as the disaster at the nuclear power plant in Japan — there would be human-assisted (tele-operated) robots that could be deployed to help out the humans in a number of specific tasks using standard tools and equipment, including a vehicle. This would be especially useful in areas where the danger was high to human rescue workers — such as collapsed buildings in war zones, terror incidents, or accidents, as well as the danger that would be found at a damaged and highly radioactive nuclear power plant.

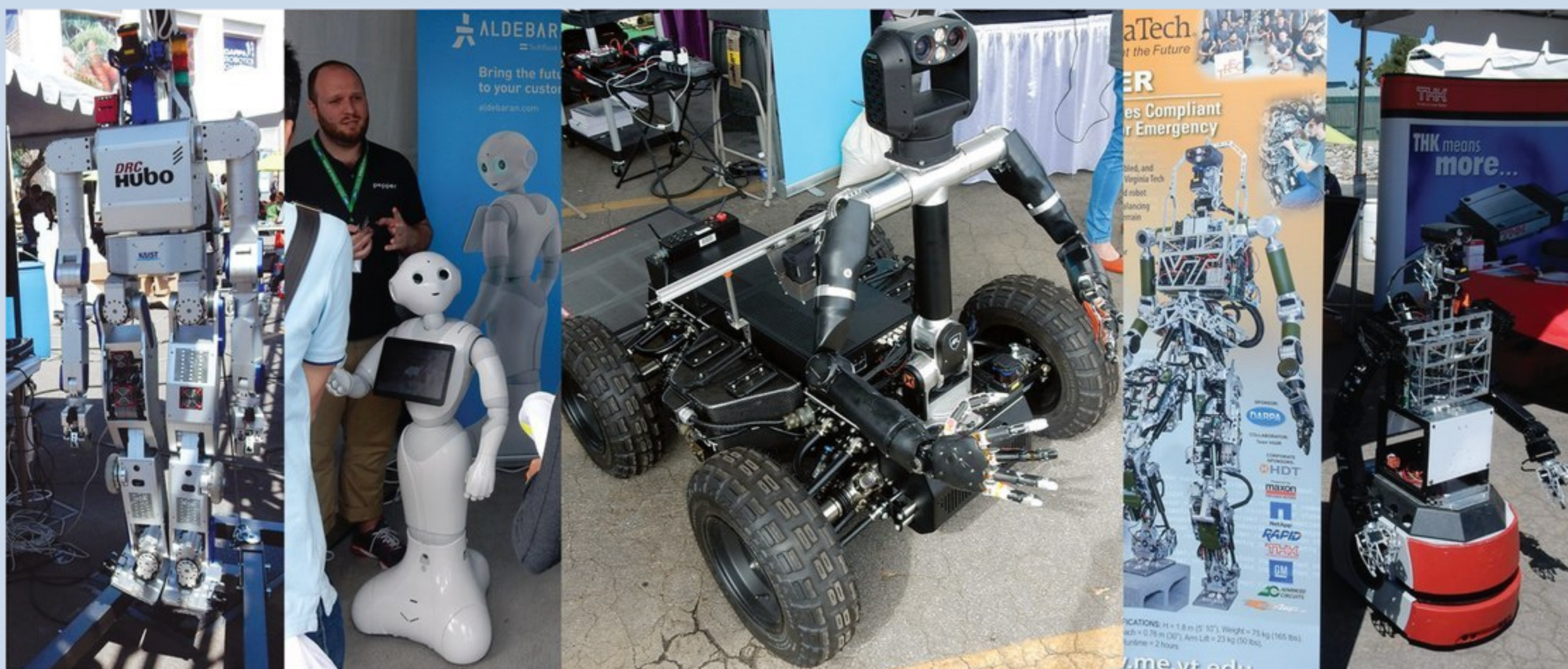
In DARPA's own words:

"The DRC Finals will require robots to attempt a circuit of consecutive physical tasks, with degraded communications between the robots and their operators; the winning team will receive a \$2 million grand prize; DARPA plans to award \$1 million to the runner-up, and \$500,000 to the third-place team."

The DRC Trials occurred in December 2013 and tasked the teams of roboticists with a set of goals similar to what would be in the Finals. To become part of the DRC Finals, a team first had to qualify unless they had competed at the Trials and had received at least eight points.

Competing robots are operated remotely, along with having some autonomous capabilities. DARPA encouraged robot autonomy by "degrading" communication which forced competitors to either wait out the lack of communication or develop autonomous software to continue during these data "brownouts."

Many of the teams did not add much autonomy, and instead waited out the long periods of degraded



communications. This was quite evident. Many robots in the competition relying on mostly human decisions and control would freeze in place for minutes at a time while waiting for better communication.

Two of the toughest parts of the competition were when the robots had to: 1. Get out of the Polaris ("Egress"); and 2. Climb the stairs. Maybe these robots should have taken a few lessons from Honda's Asimo who is quite adept at maneuvering stairs.

Tasks and Scoring

All tasks in the challenge had to be completed in 60 minutes or less. Like any competition, the DRC had rules for scoring points. Each of the below tasks when completed were worth one point:

"Outdoor" Tasks

1. Drive: Drive an open-top Polaris Ranger utility vehicle over a short course.
2. Egress: Get out of the vehicle.
3. Door: Open a door and walk through the doorway.

"Indoor" Tasks ("degraded communication")

4. Valve: Open a valve.
5. Wall: Use a drill with a rotary blade to cut a hole in drywall and remove the piece that is cut.
6. Surprise: A "surprise" task which was only revealed on the second day of the competition.
7. Rubble: "Traverse some rubble" by either crossing over a debris grade of cinder blocks, or cross a level debris field that was full of pipes and planks, pushing debris out of the way.
8. Stairs: Climb a short set of stairs.

There were penalties, as well:

If a robot falls, the robot may right itself or the human team members may set it right, but at the cost of a 10 minute penalty. There were many falls, and thus many 10 minute penalties at the event. If a robot could not for any

reason continue in any task, it could be reset and the team would lose 10 minutes, then could restart the task.

The Robots are Taking Over! Everyone Crawl for Your Lives!!!

Coming into the challenge, I knew two things: 1. What I was going to see was leading-edge robotics. No robots in the world – aside from those that had competed in the DRC Trials (and possibly Asimo) – came close to executing the tasks that were to be done; and 2. These robots generally have two speeds: very slow and stopped. (Sigh.)

Watching Team IHMC's Running Man get out of the Polaris vehicle in over three minutes really was both genuinely exciting and a bit of a letdown at the same time. I was told that many of the robots used little autonomy software, and most navigated and performed the task by tele-robotics which uses a "squishy computer" usually called "wetware" or, more commonly, "a human" to make decisions, initiate, direct, and often remotely perform most of the actions during the competition.

I watched Running Man drive his burgundy and white Polaris swiftly through the designated course only to stare in disbelief when it took about three minutes for him to exit the vehicle. I learned later that each and every step Running Man took was planned out and executed by humans, adding to the delay. One friend said watching the competition was like watching paint dry. (Sigh again.)

Cool Robot Demonstrations

There were other various robots demonstrated on the field during the event. I watched the Boston Dynamics' Cheetah robot run and jump. Unfortunately, after showing off for less than two minutes, it was put on its harness and wheeled away. I also got to see Oregon State University's Atrius robot – a biped bird-like robot that hopped from leg to leg like a child needing to go to the bathroom. I watched



as it hopped seemingly impatiently, waiting for the man with the joystick to drive it forward and then backwards a bit. Then, I was at first quite amused and then a bit upset when someone — probably one of the robot's builders — gave it a swift hard kick, knocking it back a few feet. It staggered but did not fall over.

Next, it was pelted several times with dodgeballs. It recoiled from the impact but still remained standing. I had seen that kind of thing done with the Big Dog robot, but to see this bird-like robot get kicked and pelted with dodgeballs made me feel bad for the strong and nimble steel animal.

My head knew that no animal was really being hurt, but my heart wanted to tell that bully to stop tormenting the steadfast motorized creature; to let the poor beastie be!

Notable Robots — Winners and Losers

IHMC's Running Man (an Atlas robot) — who came in second in the competition — drove its car so smoothly and so swiftly, I truly believed for that moment I was watching the genesis of the age of robotics. Of course, when it took him three and half minutes to exit the Polaris and another three minutes of painfully slow walking to get from the car to the door opening task, I was quickly disabused of that idea. Yet, everything has a beginning and with so much time, effort, and money going into robotics these days, this very well may be the beginning of the age of robotics.

Team Grit from Colorado was another impressive robot. Not in its scoring or timing — no. It caught my

	Team	Name	City	Company/ College	Score	Time
1st Place	TEAM KAIST	DRC-HUBO	Daejeon, South Korea	KAIST HuboLab & Rainbow Co.	8	44:28:00
2nd Place	TEAM IHMC ROBOTICS	Running Man	Pensacola, FL	Institute for Human and Machine Cognition	8	50:26:00
3rd Place	TARTAN RESCUE	Chimp	Pittsburg, PA	Carnegie Mellon University	8	55:15:00
	TEAM NIMBRO RESCUE	Momaro	Bonn, Germany	University of Bonn	7	34:00:00
	TEAM ROBOSIMIAN	RoboSimian	Pasadena, CA	Nasa JPL	7	47:59:00
	TEAM MIT	Helios	Cambridge, MA	Massachusetts Institute of Technology	7	50:25:00
	TEAM WPI-CMU	WARNER	Worcester, MA and Pittsburgh, PA	Worcester Polytechnic Institute and Carnegie Mellon University	7	56:06:00
	TEAM DRC-HUBO AT UNLV	Metal Rebel	Las Vegas, NV	Howard R. Hughes College of Engineering	6	57:41:00
	TEAM TRAC LABS	Hercules	Webster, TX	TRAC Labs	5	49:00:00
	TEAM AIST-NEDO	HRP2+	Ibaraki, Japan	National Institute of Advanced Industrial Science and Technology	5	52:30:00

Table 1. Top 10 DRC Teams.



attention because they only spent about \$20,000 in cash for their entry and not one person on the team was paid. They only prepared for the DRC for a very short six months, as opposed to Team IHMC who spent about three years.

Humanoid Robots — the Slow Advance of an Electro-mechanical Science

Modern humanoid robotics engineering has had several false starts — first in the 1940s with Electro, and later with the myriad of toy and hobby rolling robots that were being designed, built, and sold in the 1980s — followed by walking robots about five years ago.

The mechanics were the easy part. The programming, not so easy. The biggest barrier robotics had even 10 years ago was sheer computer processing power. With so many sensors bringing in so much data and movements happening in real time — in three dimensions — older computers could not keep up. I remember when I was working as a technician for Androbot in the mid '80s that one of the engineers told me that he watched a rolling robot take *three hours* to navigate its way around a box. Navigation, balance, finding objects, and voice recognition all depend upon fast computers.

Our modern computers might just be getting close to what is needed. Or, perhaps we just need better algorithms.

Your Robot Overlords are Here Now and Taking Over the World! If Only They Could Just Stop Falling Down ...

At the DRC, there were quadrupeds and biped robots. Walking on four legs is relatively easy — just ask any dog or baby. Walking on two legs is another story. Between 1937 and 1939, one of the first “walking” (it was more like a

shuffle) two-legged robots built was Westinghouse Electric’s “Electro.” At seven feet tall — with both voice input and output — Electro captured the imagination of science fiction writers and inventors alike. Still, fiction outpaced science and walking bipedal robots didn’t advance much until Honda’s humanoid robot development which began in the mid-1980s.

Honda, with a lot of hubris, a lot of money, and a lot of falling down (just go to honda.com and watch the early robots fall, fall, and fall again), got its robots walking. Now, its Asimo robot runs, punts a soccer ball, and climbs stairs, proving that with a “modest” budget of say a \$100 million or so, anything is possible. Now, Honda was never known for sharing its technology much, so aside from Asimo, most of the bipedal advances happened independently of the huge car maker.

In spite of all the advances in walking robots, during the challenge, there were more than 10 robots that fell over — some more than once. Many sustained damage, but were able to continue. One man described the walking robots as sure footed as toddlers. After watching a YouTube “fall compilation” video that went viral, I prefer to think of them as surefooted as someone who has had too much to drink.

Machine Intelligence or Human Intelligence?

Based on what I could see, it looked like some teams used a brute force and dumb luck method of competing, while others — particularly the college teams — used a pure math and engineering method. Based on the teams that won, I am guessing that the engineers and mathematician teams might have had a large advantage over the others.

After doing my own research, I discovered that the amount of human intervention varied greatly from team to team. Still, theverge.com had this to say about the American Team IHMC that came in second: “... (The IHMC team) focused on balancing and walking algorithms and left the rest to ‘humans.’” “All the real cognition was inside



the garage,” said Matt Johnson, one of the team leaders.

It seems like dependable autonomous walking robots are still just a science fiction dream. With so many of the robots having trouble just getting from one location to the next — and most of the navigation done tele-robotically — I guess none of these robots are going to turn into Terminators anytime soon. Skynet will just have to wait a few more years.

A College of Robotics

As the day wound down, I made my way back to my own table to promote my new college of robotics engineering. Chatting a minute with my wife, I found that we were out of flyers, and had a handful of interested students, but not one potential investor. Still, we gave out about a thousand flyers, so at least we know interest is high.

Meccanoid G15 KS: A Perfect Way to End a Perfect Day

So, with all the flyers gone, all the competitions finished, the vendors packing up, and most of the people leaving, I decided it was time to pack up Robot Betty9, my wife, my daughter, and my father-in-law and be on our way.

As fate would have it, though, I saw one more robot that I had not noticed, just standing under the Robotis tent. It was a four foot tall toy robot — a robot I had only seen before on videos of the last Consumer Electronics Show (CES). It was Meccano’s Meccanoid G15 KS — a kid-sized chattering, arm flailing, big-eyed robot, and it was right there not two feet from Robot Betty9. I stopped and stared a minute. Its owner was chatting with one of the Robotis people. I interrupted. I asked if they were for sale yet. I was told “no” and would not be for quite a few months.

I told the Robotis vendor that I had seen it on a video of the CES and that I would like to buy a pre-release so I could review it on my **robots-and-robots.com** website. He said that I could and if I followed him to his truck he

would give me one. As we walked to his car, I offered up my credit card. He turned it down and said I could just have it if I was going to review it. I said I was, and I smiled broadly as we got to his truck. He reached in, grabbed a Meccano’s Meccanoid G15 KS kit from a stack of about 10, and put it in my hands. He then drove me to my car where I found my wife and her father waiting, and my nine year old daughter in her R2D2 jacket looking at me with eyes as big as saucers, wondering what that big box with the robot’s picture on the front contained.

I thanked the man as he drove off, putting the Meccanoid box into my daughter’s arms. I explained it was a robot kit as she looked at the front, sides, and back of the box. It was a bit heavy for her, so she put it down and asked, “Is it mine?” “It’s ours,” I answered. “We’ll build it together and then I’ll write a review on it.” “Oh, Daddy!” she replied with a toothy ear-to-ear smile. We had built many robot kits together before this, but none this massive, this tall, or this impressive.

We packed the robot kit into the trunk, my daughter got in the car, and I sat Robot Betty9 on my daughter’s lap (she insisted), then started heading home. We would be building that robot for many evenings. I thought ahead to the next competition — NASA’s space robotics challenge — more fun was yet to come. **SV**



Photo by Thomas Messerschmidt

The *SERVO* Webstore

CD-ROM SPECIALS

**11 CD-ROMs
&
Hat Special**



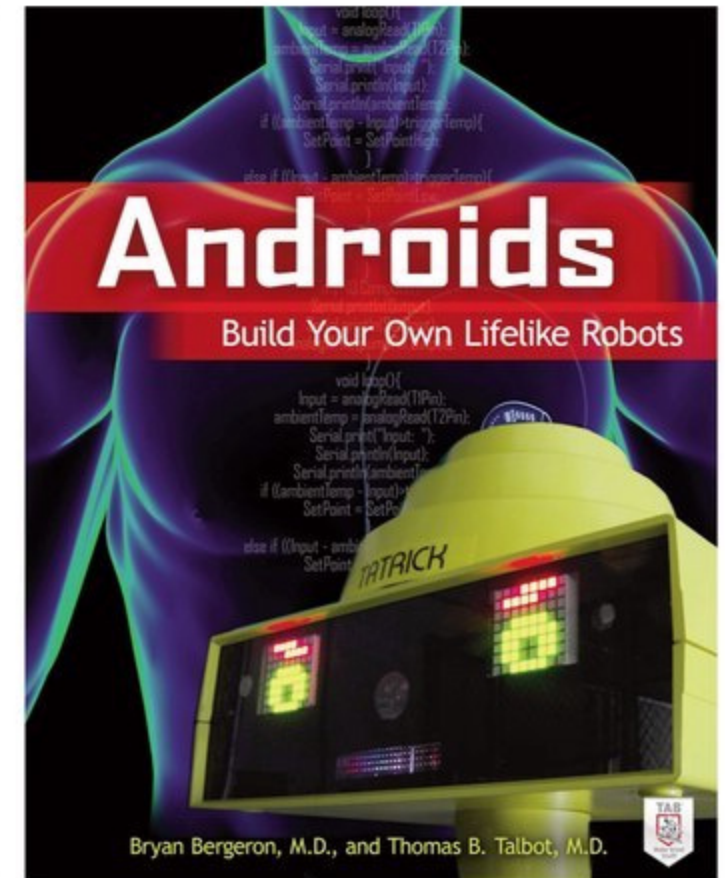
Free Shipping!

**Only \$ 229.95
or \$24.95 each.**

8.2015 FEATURED

Androids

Build Your Own Lifelike Robots



Bryan Bergeron, M.D., and Thomas B. Talbot, M.D.

ROBOTICS

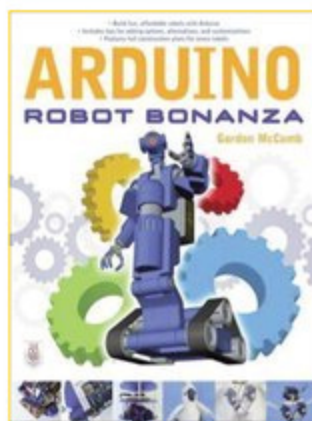
Arduino Robot Bonanza

by Gordon McComb

Create high-tech walking, talking, and thinking robots. "McComb hasn't missed a beat. It's an absolute winner!"

Breathe life into the robots of your dreams — without advanced electronics or programming skills. *Arduino Robot Bonanza* shows you how to build autonomous robots using ordinary tools and common parts. Learn how to wire things up, program your robot's brain, and add your own unique flair.

\$29.95



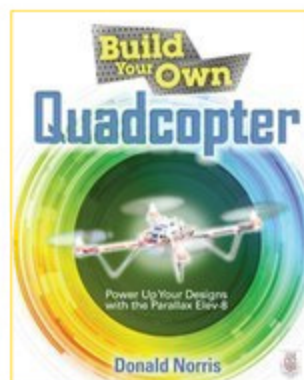
Build Your Own Quadcopter: Power Up Your Designs with the Parallax Elev-8

by Donald Norris

Build a custom multirotor aircraft!

Build and customize radio-controlled quadcopters that take off, land, hover, and soar. *Build Your Own Quadcopter: Power Up Your Designs with the Parallax Elev-8* features step-by-step assembly plans and experiments that will have you launching fully functioning quadcopters in no time.

\$50.00

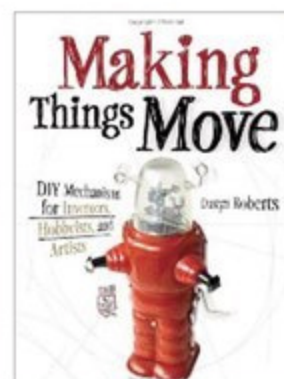


Making Things Move: DIY Mechanisms for Inventors, Hobbyists, and Artists

by Dustyn Roberts

In *Making Things Move: DIY Mechanisms for Inventors, Hobbyists, and Artists*, you'll learn how to successfully build moving mechanisms through non-technical explanations, examples, and do-it-yourself projects — from kinetic art installations to creative toys to energy-harvesting devices. Photographs, illustrations, screenshots, and images of 3D models are included for each project.

\$29.95



Build Your Own Humanoid Robots

by Karl Williams

GREAT 'DROIDS, INDEED!

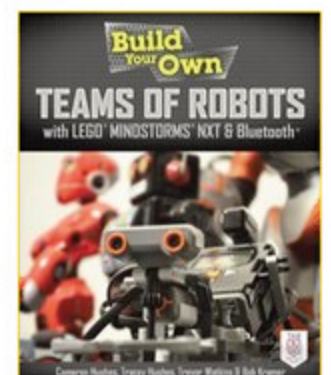
This unique guide to sophisticated robotics projects brings humanoid robot construction home to the hobbyist. Written by a well-known figure in the robotics community, *Build Your Own Humanoid Robots* provides step-by-step directions for six exciting projects, each costing less than \$300. Together, they form the essential ingredients for making your own humanoid robot.

\$24.95



Build Your Own Teams of Robots with LEGO® MINDSTORMS® NXT and Bluetooth®

Plan, design, assemble, and program robot squads that communicate and cooperate with each other to accomplish together what they can't do individually. Inside the authors show you how to construct a team capability matrix (TCM) and use the Bluetooth Robotic-Oriented Network (BRON) so your robot teams can share sensors, actuators, end effectors, motor power, and programs. **\$30.00**



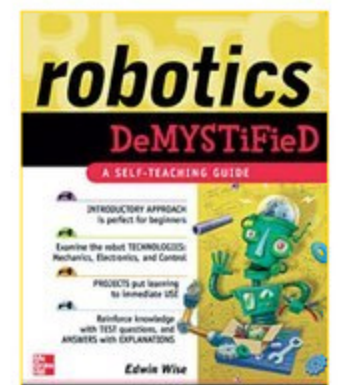
Robotics Demystified

by Edwin Wise

YOU DON'T NEED ARTIFICIAL INTELLIGENCE TO LEARN ROBOTICS!

Now anyone with an interest in robotics can gain a deeper understanding — without formal training, unlimited time, or a genius IQ. In *Robotics Demystified*, expert robot builder and

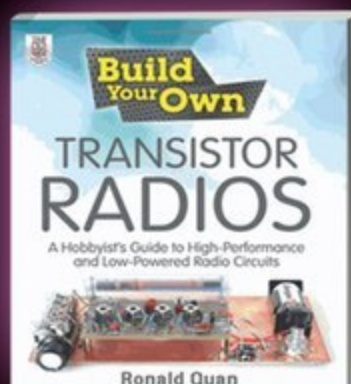
author Edwin Wise provides an effective and totally painless way to learn about the technologies used to build robots! **\$19.95**



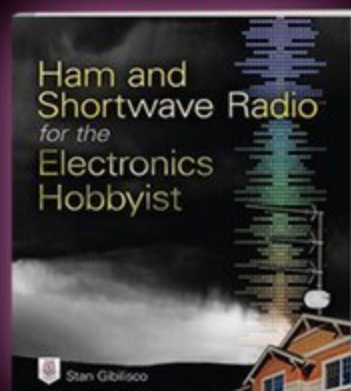
**We accept VISA, MC, AMEX,
and DISCOVER**
Prices do not include shipping and
may be subject to change.

SPECIAL OFFERS!

DO YOU LOVE RADIOS?



**Sale Price
\$39.95**



**Sale Price
\$19.95**



**Sale Price
\$23.95**

Get 20% off these three books

To order call 800 783-4624

PLUS

Free Std. Shipping!

Limited Time Only

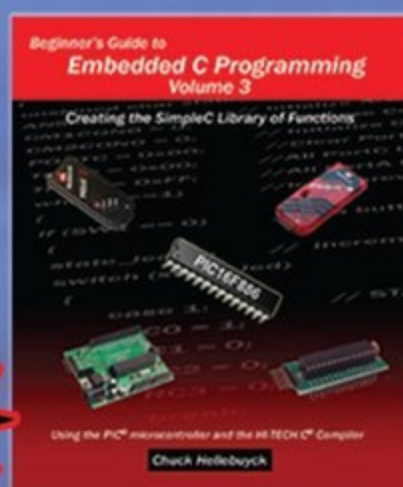
Forbidden LEGO

by Ulrik Pilegaard / Mike Dooley
Forbidden LEGO introduces you to the type of free-style building that LEGO's master builders do for fun in the back room. Using LEGO bricks in combination with common household materials (from rubber bands and glue to plastic spoons and ping-pong balls) along with some very unorthodox building techniques, you'll learn to create working models that LEGO would never endorse.



Reg \$24.95 Sale Price \$19.95

Beginners Guide Book Combo.



Only \$85.95

**Plus
FREE Priority Mail Shipping
US Only**

"THE PHANTOM DRAW"

The KILL A WATT meter is the best way to help you determine your actual energy draw in ON and OFF home appliances.



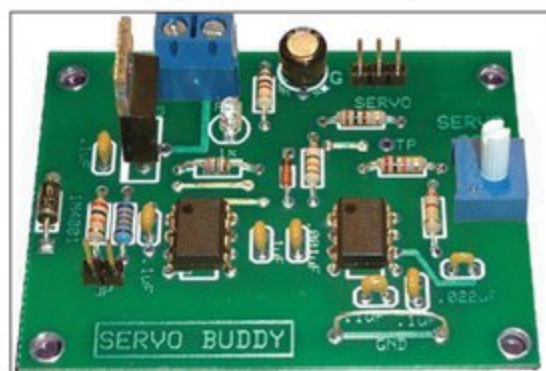
Only \$29.95

To order call

1 800 783-4624 or online www.servomagazine.com

PROJECTS

The SERVO Buddy Kit



An inexpensive circuit you can build to control a servo without a microcontroller.



For more information, please check out the May 2008 issue or go to the SERVO webstore.

Includes an article reprint.
\$39.55

3D LED Cube Kit

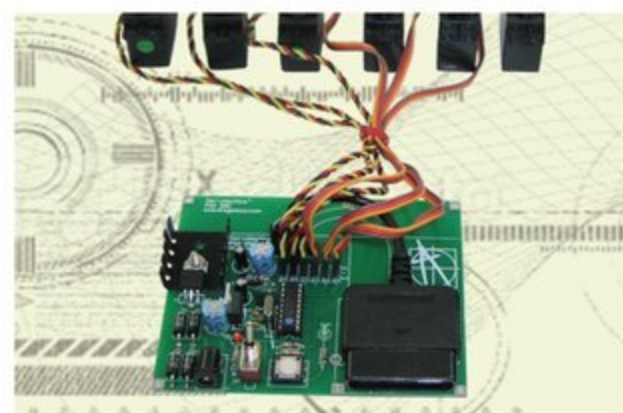
From the article "Build the 3D LED Matrix Cube" as seen in the August 2011 issue of *Nuts & Volts Magazine*.



This kit shows you how to build a really cool 3D cube with a 4 x 4 x 4 monochromatic LED matrix which has a total of 64 LEDs. The preprogrammed microcontroller that includes 29 patterns that will automatically play with a runtime of approximately 6-1/2 minutes. Colors available: Green, Red, Yellow & Blue. Jig and plastic cases also available.

\$57.95

PS2 Servomotor Controller Kit



This kit accompanied with your own PlayStation controller will allow you to control up to six servomotors. Includes all components and instruction manual. For more information, please see the February 2011 edition of *SERVO Magazine*. Assembled units available!

\$79.95

To order call 1-800-783-4624
Or Online At www.servomagazine.com

SERVO Magazine Bundles



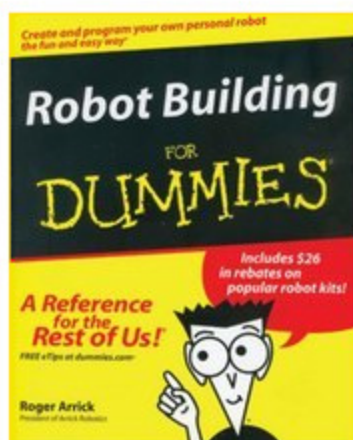
**Save \$10.00
Only \$57.95!**

Now you can get one year's worth of all your favorite articles from *SERVO Magazine* in a convenient bundle of print copies. Available for years 04, 05, 06, 07, 08, 09, 10, 11 and 2012.

Robot Building for Dummies

by Roger Arrick / Nancy Stevenson

Discover what robots can do and how they work. Find out how to build your own robot and program it to perform tasks. Ready to enter the robot world? This book is your passport! It walks you through building your very own little metal assistant from a kit, dressing it up, giving it a brain, programming it to do things, even making it talk. Along the way, you'll gather some tidbits about robot history, enthusiasts groups, and more.



\$24.95

*Any bot builders
out there?
Get cool
robotics stuff
from my store!*

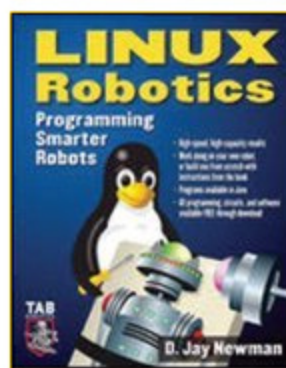
*Call me at my
order desk!*

Visit my online store @
www.servomagazine.com

Linux Robotics

by D. Jay Newman

If you want your robot to have more brains than microcontrollers can deliver — if you want a truly intelligent, high-capability robot — everything you need is right here. *Linux Robotics* gives you step-by-step directions for “Zeppo,” a super-smart, single-board-powered robot that can be built by any hobbyist. You also get complete instructions for incorporating Linux single boards into your own unique robotic designs. No programming experience is required. This book includes access to all the downloadable programs you need.



\$38.95

CNC Machining Handbook: Building, Programming, and Implementation

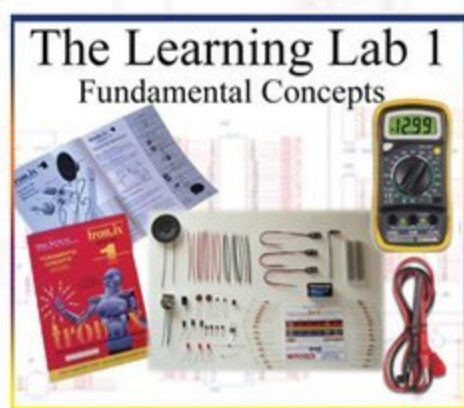
by Alan Overby

The *CNC Machining Handbook* describes the steps involved in building a CNC machine and successfully implementing it in a real world application. Helpful photos and illustrations are featured throughout. Whether you're a student, hobbyist, or business owner looking to move from a manual manufacturing process to the accuracy and repeatability of what CNC has to offer, you'll benefit from the in-depth information in this comprehensive resource.

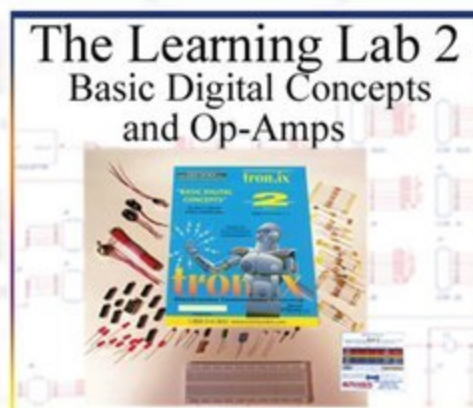


\$34.95

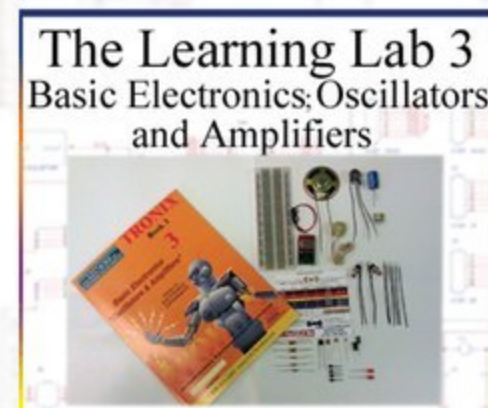
LEARN ELECTRONICS - ESSENTIALS FOR BEGINNING BOT BUILDERS



\$59.95



\$49.95



\$39.95

These labs from LF Components show simple and interesting experiments and lessons, all done on a solderless circuit board. As you do each experiment, you learn how basic components work in a circuit, and continue to build your arsenal of knowledge with each successive experiment.

For more info please visit our webstore.



MAXROBOTTECH™

BY: AARON & ROBERT GROSS

BEHAVIORS, MAPS, LANDMARKS

Robot intelligence doesn't have to be complicated. Landmarks allow robots to know where they are. The stairs are an easily recognized landmark.

With simple programming, Albert doesn't care that it's a new room, and gets moving. Using simple rules, called behaviors, Albert moves with minimal processing.

Albert found the stairs but won't remember where the stairs are. Even so, he knows that they are visible and will use movement and sensors to look for them every time. (For simple areas, this works well because if the robot cannot see the landmark it has to move anyway.)

Mapping initially slows JJ and BOB, but detailed maps will later allow them to move directly from place to place. They will place the landmark on an internal map.

Programming the amount and type of intelligence your robot needs is part of the fun of robotics. Simple robot code can be very effective.

© 2015

www.maxrobottech.com

The Adventures of Bogie and Sprout

It's a great time to be a roboticist. With so many microcontroller board options like Arduino, Raspberry Pi, and SparkFun RedBoard, it's never been easier to find a brain for your robot. However, a brain without a good driving base is like Batman without Robin, or Frodo Baggins without Samwise Gamgee. The folks at Actobotics have the perfect solution for those looking for a driving base that is easy to build, ripe for expansion, and super affordable: the Runt Rovers.

Will the Runt Rovers be able to save the day for roboticists stymied by the lack of a good driving base for their microcontroller boards? With the help of two of the Runt Rovers — the simple two-wheeled Sprout and the more sophisticated six-wheeled Bogie — we were sure to find out.

Everyone Loves an Origin Story

The Runt Rovers are an offshoot of the Actobotics building system and are available through ServoCity. We had our first exposure to Actobotics last time when we worked with the system in conjunction with the Force servo arm. The precision machined aluminum and distinctive hole pattern gives the parts a stylish polish and a cool factor that usually seems to evade plastic parts. The hole pattern makes integrating servos into your designs so much easier.

With other kits, the small unusually spaced holes on servo horns would rarely match up with larger structural parts. The hole pattern on Actobotics parts are the perfect fit for just about any servo attachment you could imagine, but Actobotics is perfect for so much more than servos. The system also features great parts like mounting brackets, hub mounts, gears, and sprockets for use with motors, too.

After having such a positive first experience, we had high hopes for the Runt Rovers. One difference between the Runt Rovers and a lot of the other Actobotics parts is immediately apparent. While they still sport some of the familiar hole patterns, the structural parts for the Runt

Rovers are all plastic. This is not at all meant as a knock against the kits. In fact, the thrifty construction is one of their greatest virtues. The Runt Rovers are a lot gentler on your wallet than slobbery Hooch to the pristine interior of Turner's car. The six different Runt Rovers range in price from the bargain rate of about \$16 for the simplest Runt, to the still very bargain price of about \$70 for the most sophisticated Rover.

The six Runt Rovers are a veritable Justice League of efficient robot design. The simplest kit is appropriately named the Peewee. The smallest Runt (and we mean that — the website shows a Coke™ can towering over it for scale) has two wheels and two motors, and a simple rectangular frame. The frame includes the distinctive hole pattern in the center of the top body plate for expansion with other Actobotics parts.

The next Runt is the Sprout, which is one of the ones that we got our hands on. The Sprout is also a two-wheeler, but with a slightly larger frame and a few more mounting points. One of the great things about the ServoCity website is that on each page dedicated to the Runt Rovers, the entire kit is shown laid out with the parts identified so you know exactly what you'll be getting.

The pages also feature tons of pictures with the Runts in action, and some cool design suggestions with some soda cans for scale. Just in case you can't tell from the pictures of the complete bot, if there's one more set of mounting holes on the bottom plate, the laid out parts give you your answer.

The third Runt is Junior, which includes four wheels, four motors, and a slightly bigger frame. The fourth Runt is the Whippersnapper, which is also a four-wheeler. The Whippersnapper has a taller profile than Junior — the main body is big enough to house a standard servo sitting upright. The fifth Runt is the boxy Half-Pint.

If the Runt Rovers were the Fantastic Four, Half-Pint would be The Thing. The hefty bot provides enclosed storage for all of your electronics, and numerous surfaces

for mounting mechanisms and sensors.

The newest Runt Rover is the impressive Bogie, which is the other Runt we acquired for experimenting. The Bogie has six motors, six wheels, and a super cool rocker-bogie suspension arrangement. The Bogie towers over its fellow Runts with nearly five inches of ground clearance, and features a simple plate for mounting.

Runt Rovers Assemble

Now that we've become acquainted with the whole crew, we can take a deeper dive while we assemble the Sprout and the Bogie. First up was the Sprout. For starters, we laid out all of the parts much like is shown on the website. It's a surprisingly small number of parts — only 13 individual pieces! Perhaps most surprising is that the Sprout actually requires no screws whatsoever — everything snaps together.

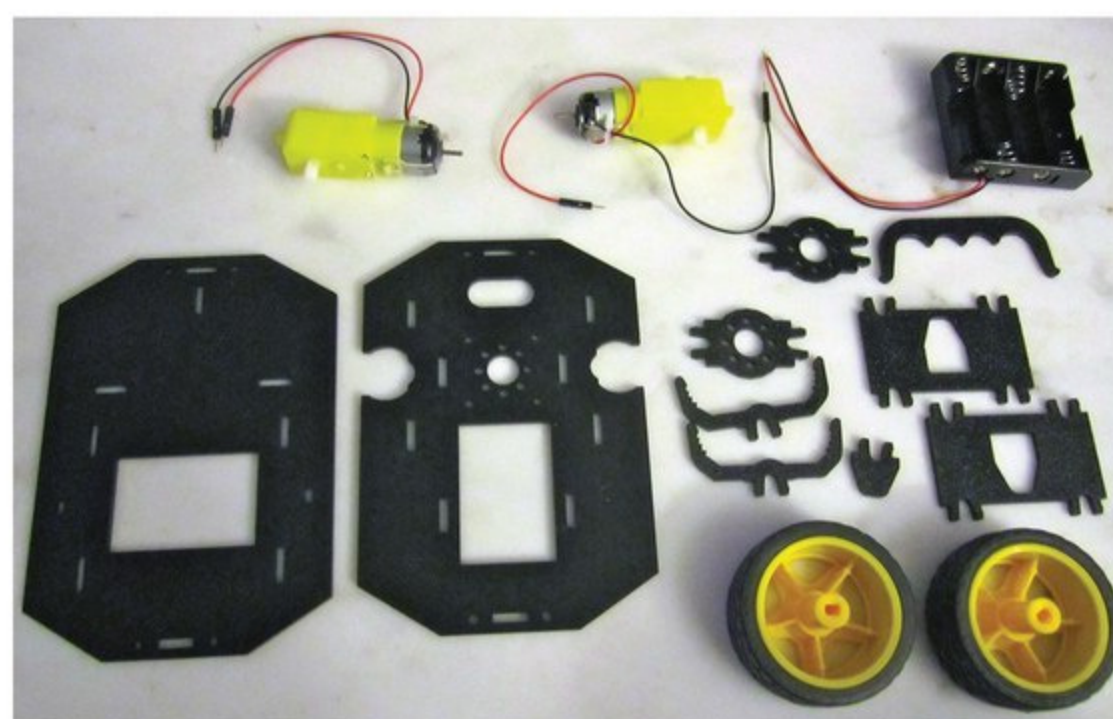
Just like with the grippers we worked with last time, the assembly instructions took the form of online videos. The familiar voice of the patient narrator takes you through each step, first identifying the parts you need and then showing exactly how things go together. The plastic parts all come with a rough edge and a smooth edge, so it's easy to know if you have them in the right orientation. The snap-together parts strike the perfect balance between being easy to put together but snug enough to assuage any worries about the bot coming apart at the seams. It's the perfect interplay, like between Johnny Carson and Ed McMahon.

Even the motors snap into the frame. The DC motors themselves are pretty impressive little packages. The motors are small and have a nice little gearbox attached to them. The output shaft from the gearbox is plastic with a flat edge, which fits nicely into the wheels that come with the kit. The motors even come pre-wired with nice solid pins on the end — not some frayed end of stranded wire. When it's all said and done, the Sprout weighs in at just over half a pound.

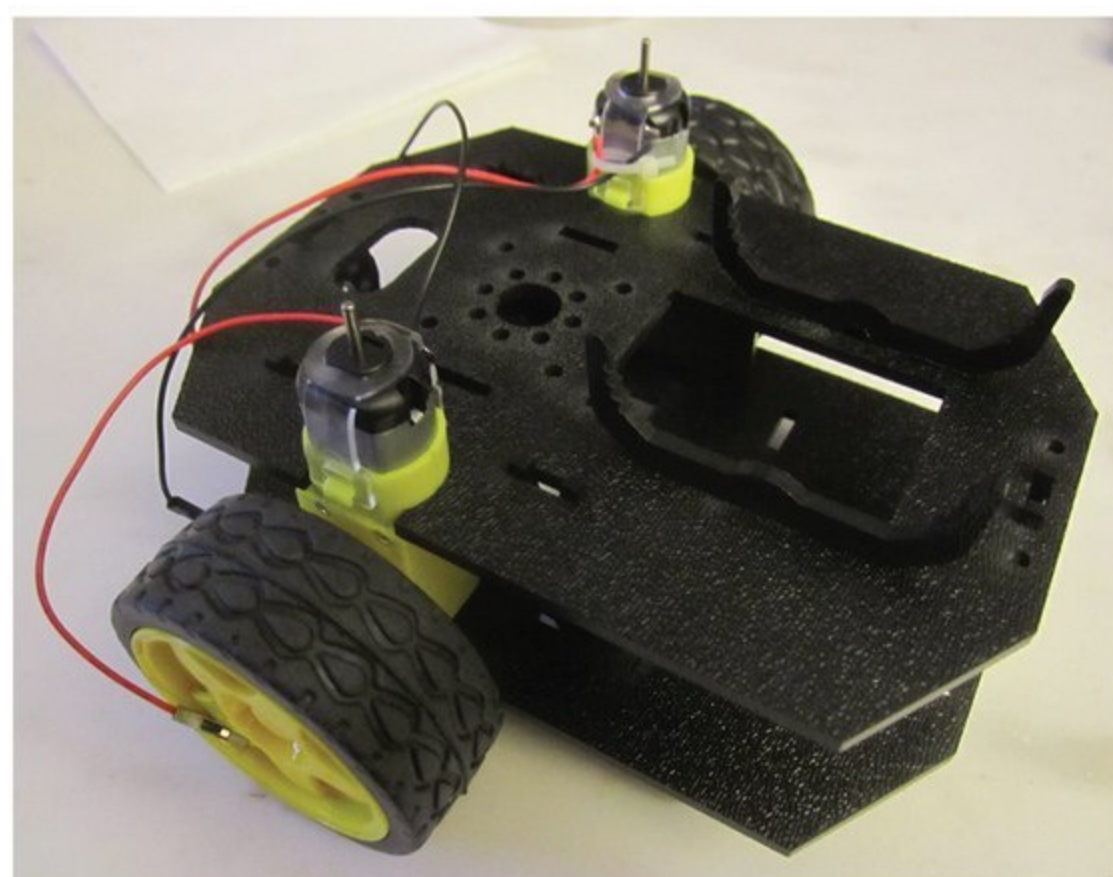
One of the features of the Runt Rovers that are touted throughout the ServoCity website are the snap-on microcontroller mounts. The mounts are two U-shaped brackets with a stepped tooth-like structure on the inside curve. They looked simple enough, but the site ballyhooed them as an innovative and hassle free way to mount any of the popular microcontroller boards.

After testing them out ourselves, we are definitely believers. We slid an Arduino Uno in nice and easy by gently spreading the pliable plastic pieces. Once in place, the board was as solidly fixated as Lippy the Lion on one of his get rich quick schemes (despite the entreaties of Hardy Har Har). We could pick up and shake the Sprout, and the Uno board was at no risk of coming loose.

The unobtrusive mount also ensured easy access to every port and connector on the board. If we wanted to switch out the Uno for something bigger like a Raspberry Pi



THE SEEDS OF THE SPROUT.



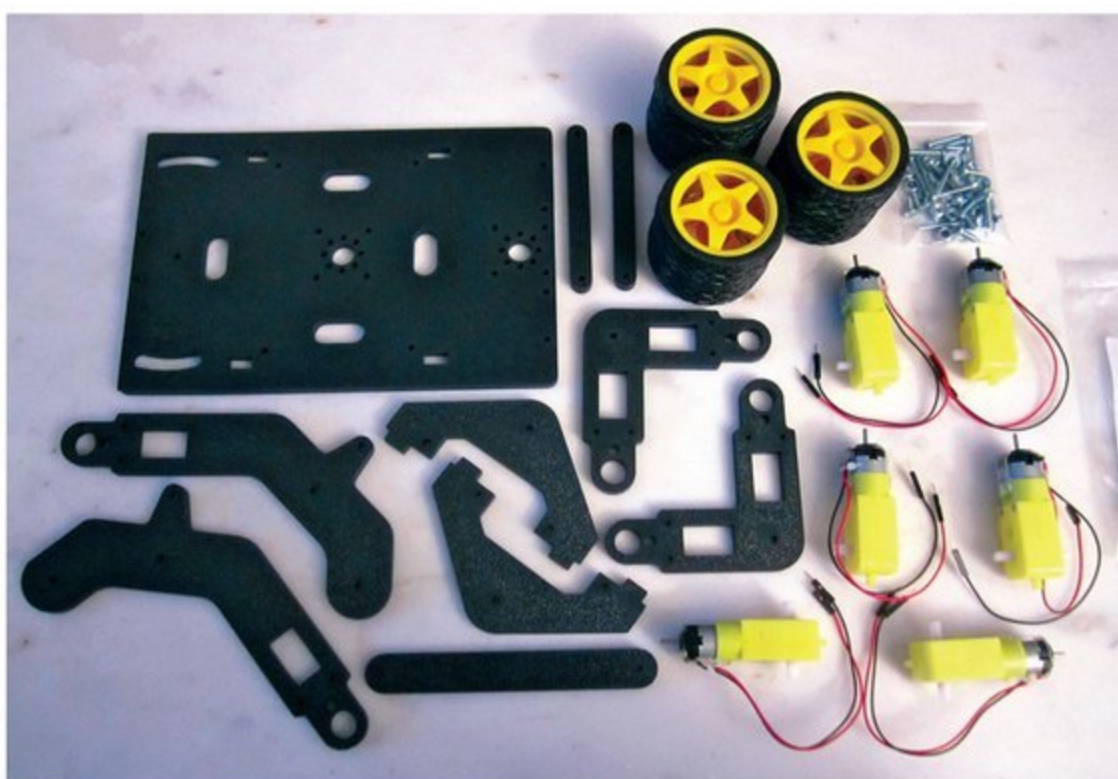
THE SPROUT COMES TOGETHER IN A SNAP.

board, all we had to do was slide it out and affix the bigger board to the wider set of teeth. No more fiddling with posts and screws whenever you want to remove or install your board.

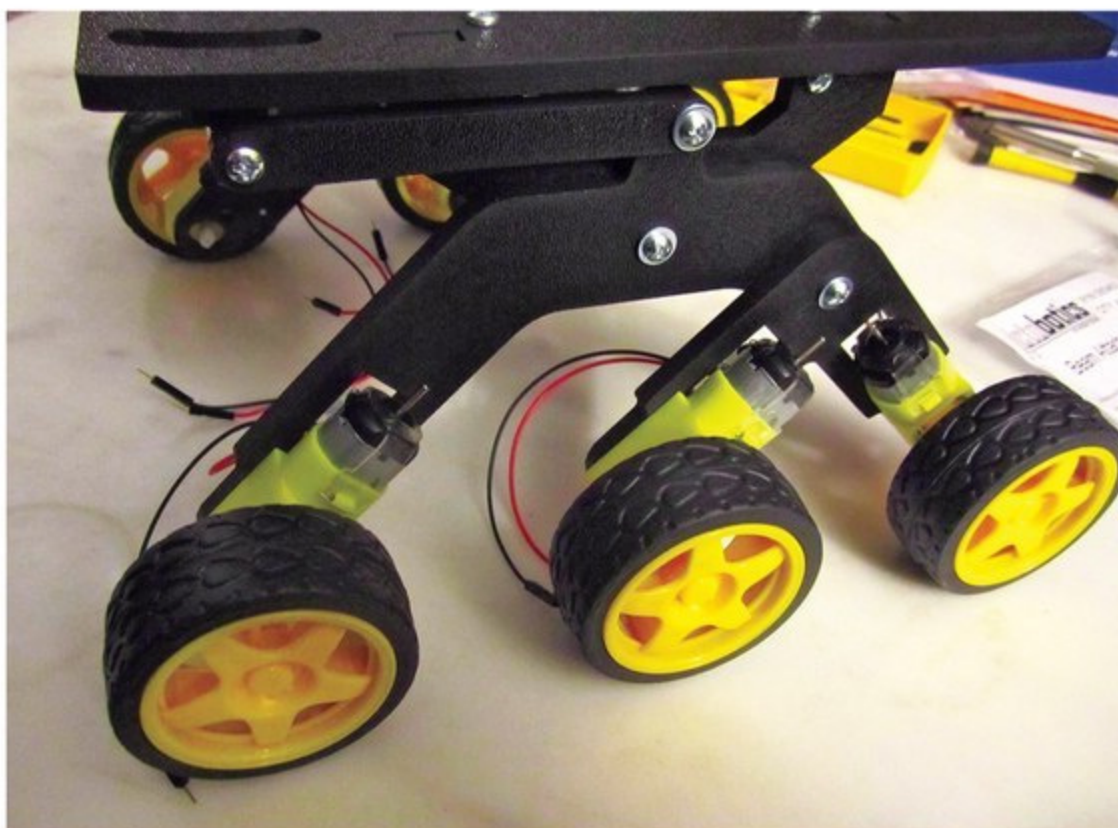
Once the Sprout was together, we were ready for the more sophisticated Bogie. The Bogie requires a few more parts, and entails screws as well as snap-together parts. At the beginning of the instructional video, it tells you what tools you need. For the Bogie, the video recommends a Phillips head screwdriver and a pair of pliers. The recommendation for pliers seemed a bit odd to us.

In the instructions, the pliers are used to hold nuts in place while the screws are tightened. A pair of pliers will indeed get the job done, but we preferred to go with a small socket wrench. Since we had the ideal tools at our disposal, we might as well use them.

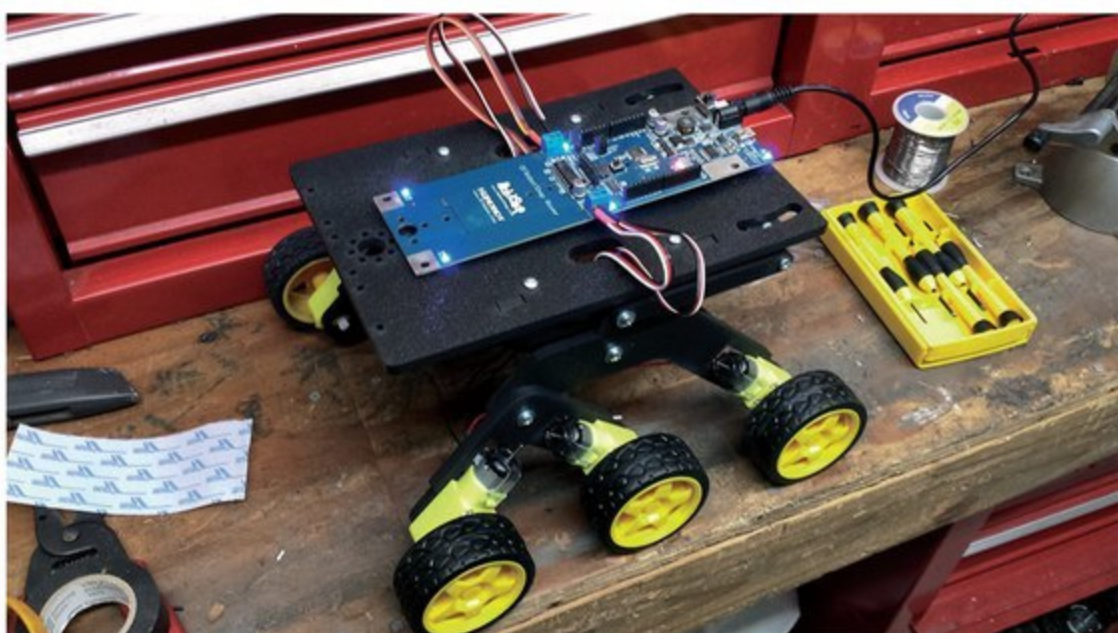
We actually think the pliers recommendation speaks to the admirable goals of the Runt Rovers. The Runts are



BOGIE PARTS.



THE ROCKER-BOGIE SUSPENSION.



IT'S ALIVE!

SPECIAL THANKS

Lauren Lewis from RobotZone



WIRING UP THE BOGIE MOTORS.

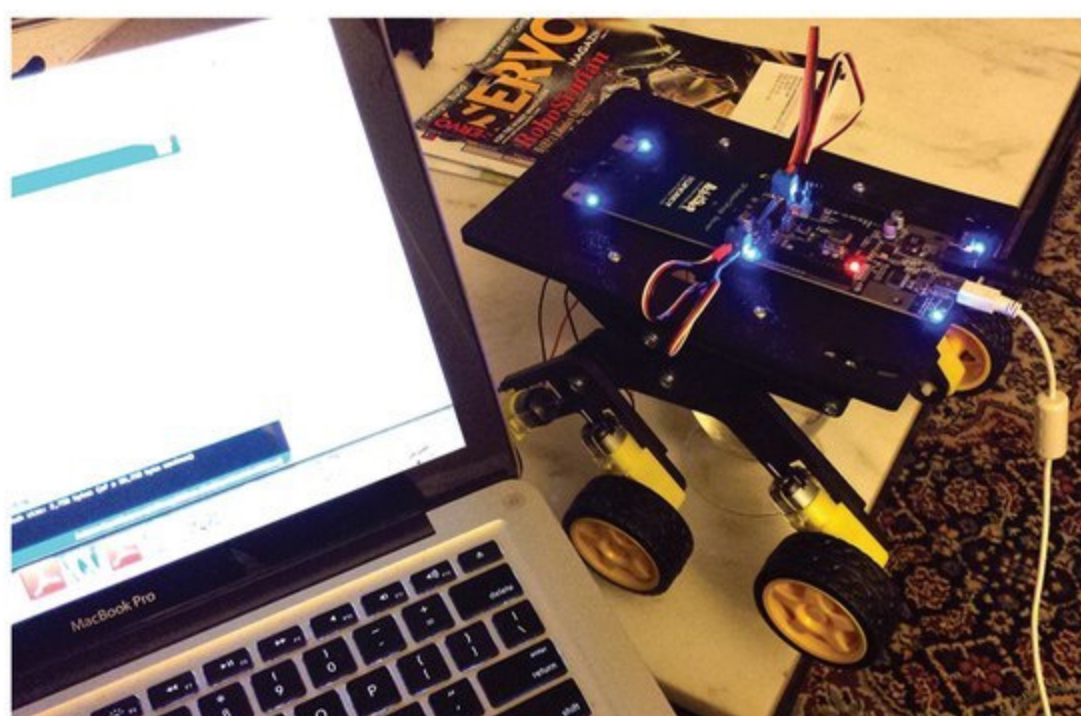
meant to be accessible robot kits that would be ideally suited to kids and students just starting out in their robotics endeavors. We think pliers are recommended because they are the type of tool that pretty much everyone will have lying around. Socket wrenches are a little more specialized, and not everyone might have a set of them on hand. In any event, pliers will definitely get the job done.

Much of the construction on the Bogie is dedicated to the cool rocker-bogie suspension from which the robot takes its name. The rocker-bogie design has quite the design pedigree — it's the preferred suspension setup of the Mars rovers including Spirit, Opportunity, and Curiosity. The rocker half of the name refers to the linkages that abide by the definition of "rocker" with respect to four bar linkages.

A rocker linkage is one that has a limited rotational range. The bogie aspect of the name refers to the driven wheels at the end of each linkage. For the Mars rovers, the rocker-bogie suspension is designed to allow the bot to tackle obstacles while keeping all six wheels on the ground, and minimizing the tilt of the main body. We love that the inclusion of the suspension design is as instructive as Navi is for Link — it's a great example of good mechanical design that provides a springboard for discussion of some of the coolest, most out of this world robots of recent memory. We couldn't think of a better way to get novice roboticists excited about the possibilities of science and engineering.

As with any truly elegant design, the rocker-bogie suspension comes together pretty easily. The linkages are connected with a few screws, and the differential between the two sides simply uses a few L-shaped brackets. The decent size of the Bogie also ensures that you're never scratching your head about how to get your screwdriver into a snug spot to tighten that last stubborn screw that seems to be as intent on hiding as Don Quixote is at tilting at windmills (despite Sancho Panza's puzzlement).

Even without wiring up the Bogie, it's fun to test out by pushing it over uneven terrain and seeing the rockers undulate. Though, of course, it would be even more fun to see it do that under its own power.



A WEE BIT OF PROGRAMMING.

Exercising the Little Grey Cells and the Battery Cells

Like the Scarecrow off to see the Wizard, all our bots longed for now was a brain. For the Bogie, we had the perfect fit. We pilfered the board from the DFRobotShop Rover, which last made an appearance in our column back in March 2011. The DFRobotShop Rover includes a board based on the Arduino Duemilanove, but what we were most excited about were the built-in motor drivers. Unfortunately, there were only two motor drivers while the Bogie had six wheels.

This conundrum, however, wouldn't require the likes of Sherlock and Watson to figure out. We resolved to use one motor driver per side, so all we would have to do is splice all of the motor wires together to a common power lead and a common ground lead. Since the end of the Runt Rover wires are pins instead of simply wire, we were able to use PWM connectors to gather them together.

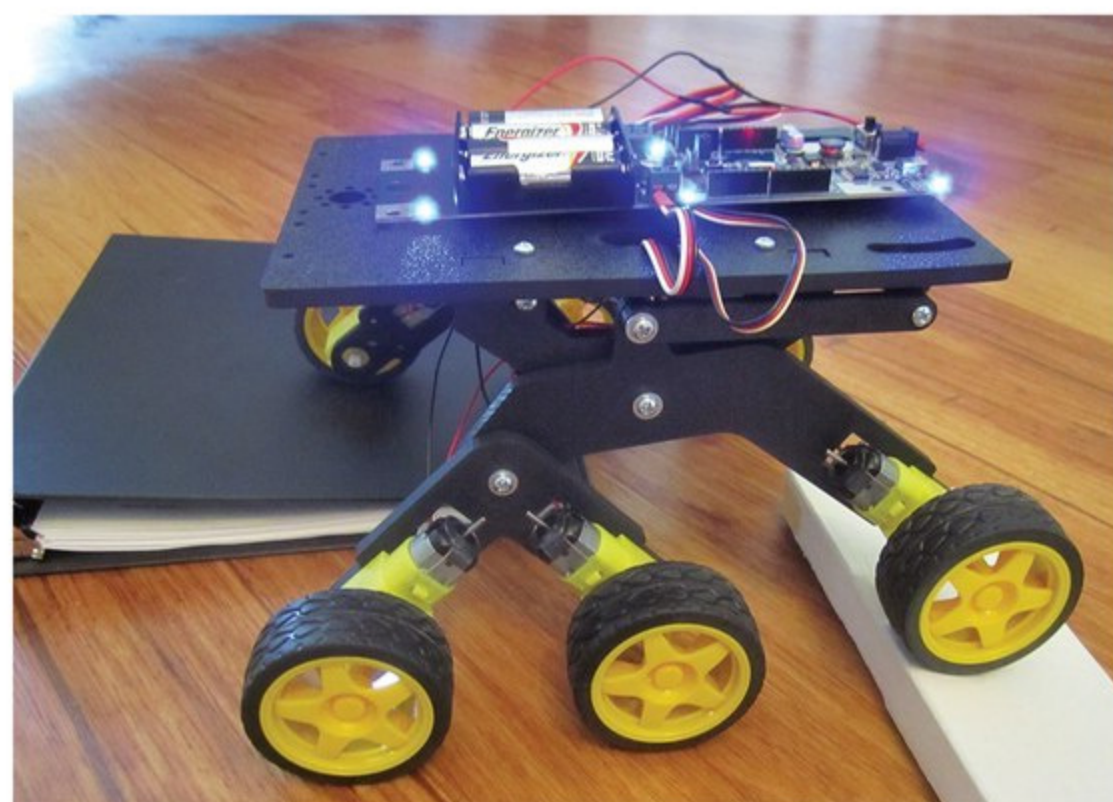
We cut the other end off of the PWM cables and simply spliced the three PWM leads together and tinned the tip. We used black or red heat shrink on the end of the wire to keep track of whether it was the common power or common ground for the side. We always prefer a solution like this that involves connectors instead of the slightly more permanent and messy solution of soldering motors. In the future, we might want to wire up the Bogie with a board that has six motor drivers — one for each motor. When that time comes, all we'll have to do is disconnect our modified PWM cables and put the motor pins directly into the drivers.

With just that minimum amount of soldering, we were ready to fire up the Bogie. Upon plugging it into wall power, the LEDs lit up with vibrant blue hues.

The Sprout is ostensibly even easier to wire up with just two motors. A board with only two motor drivers would be perfect. Here is where we come to perhaps the biggest caveat of the Runt Rovers — you need to make sure you



READY TO TACKLE TERRAIN.

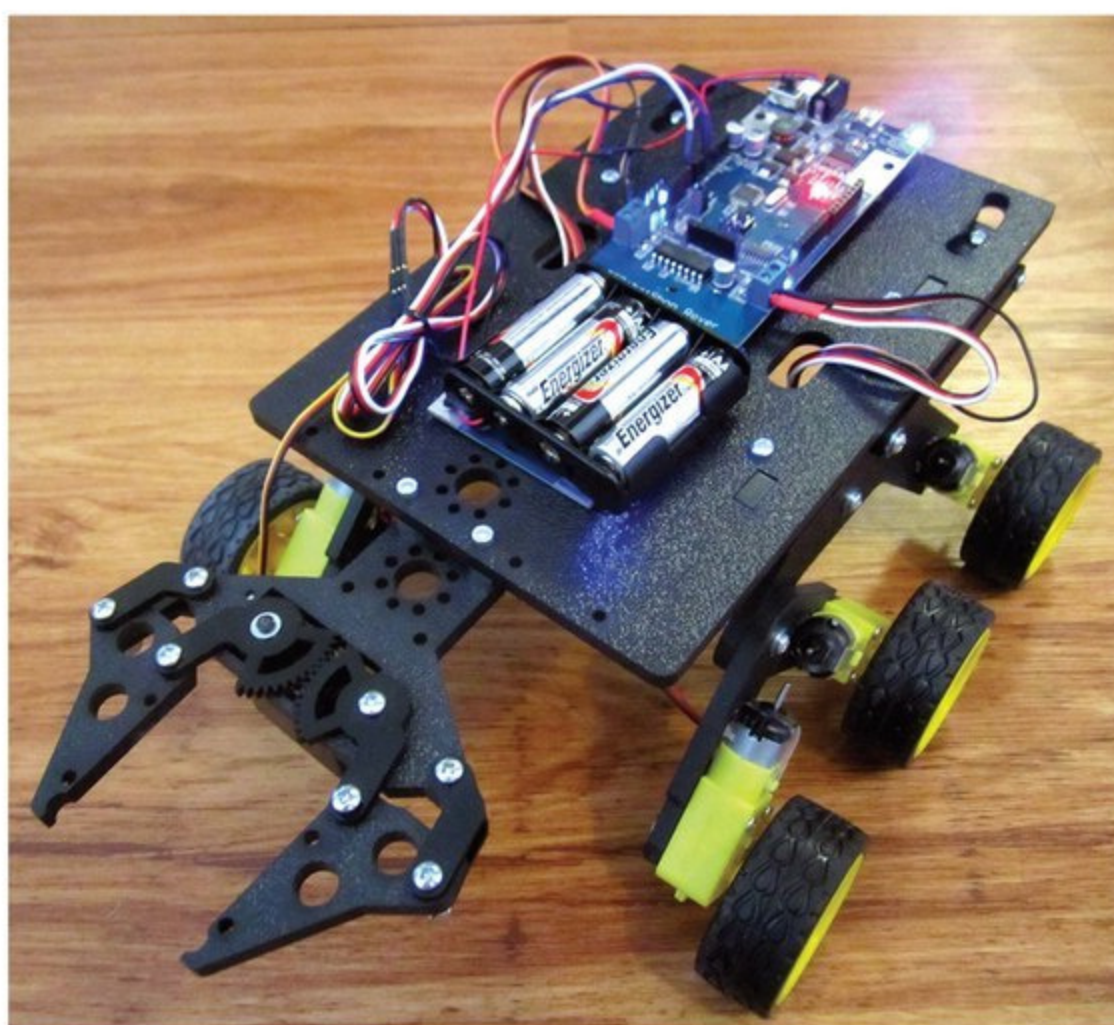


ROCKER ON.

have motor drivers to make them work, and the basic Arduino board like an Uno is not going to cut it. The Uno has plenty of analog and digital sensor ports, but no motor drivers. If the Runt Rovers were driven by servos, you could just wire them into the power, ground, and PWM ports on the Uno. However, the motors have nowhere to go, like Jem and the rest of the Holograms after a concert is cancelled.

It is theoretically possible to run some DC motors straight from an Arduino Uno board without motor drivers. If you hook both motor pins to PWM ports on the Uno, you can achieve simple motor control but this technique is risky. The maximum current rating on the I/O pins is 40 mA. You'll have to be using a tiny motor indeed for that to be enough juice.

The DC motors that come with the Runt Rovers draw 190 mA at no load and 250 mA at max load — far more than the Uno I/O pins can handle. So, by risky we mean trying to use PWM ports on the Arduino Uno to act as substitute motor drivers is a surefire way to destroy your pins.



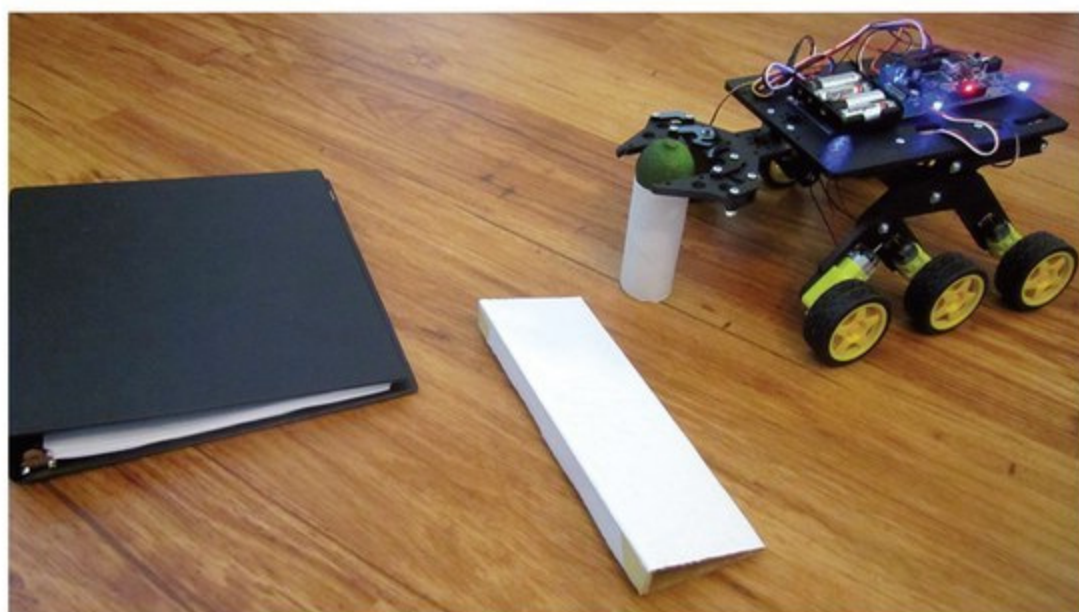
NOW WITH MORE CLAWS.

Fortunately, there's an easy solution, but it's one you'll want to have in mind when doing your shopping in the first place — get a motor driver. You could rig up some H-bridges on a breadboard if you wanted to MacGyver it, but the easiest way to go would be to get a motor shield for your board.

The great thing about boards like the Uno is that they have a lot of expandable shields for additional functionality. The motor shield will only run you around \$10 or so, so the additional part won't break the bank. Just be sure to order one in the first place, or you might be left with a Runt as helpless as Wallace without Gromit.

That was actually the precise situation we were in — we failed to get a motor driver when we got our Uno. So, our Runts would just have to share the DFRobotShop Rover Duemilanove-based board.

Programming can be as puzzling as the mysteries best



NOW WITH MORE LIMES.

```

driving_test_servo | Arduino 1.0.5

driving_test_servo $
#include <servo.h>

int E1 = 6; //M1 Speed Control
int E2 = 5; //M2 Speed Control
int M1 = 8; //M1 Direction Control
int M2 = 7; //M2 Direction Control

Servo servo1; Servo servo2;

void setup()
{
  int i;
  for(i=5;i<=8;i++)
    pinMode(i, OUTPUT);
  servo1.attach(14); //analog pin 0
  Serial.begin(9600);
}

void loop()
{
  int leftspeed = 255; //255 is maximum speed
  int rightspeed = 255;
  analogWrite (E1,255);
  digitalWrite (M1,HIGH);
  analogWrite (E2,255);
  digitalWrite (M2,HIGH);
  servo1.write(45);
  delay(5000);
  analogWrite (E1,255);
}

```

NOW WITH MORE CODE.

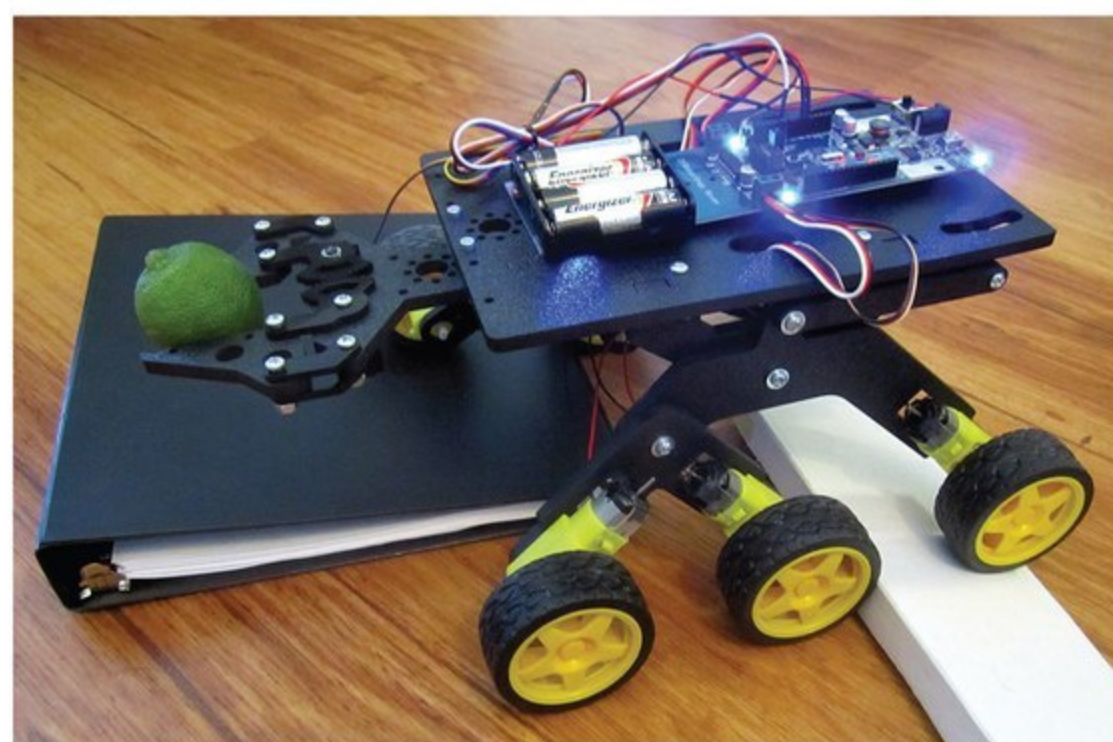
and support for Arduino is really fantastic, with plenty of pages of sample programs that are always annotated with links to any other required libraries.

The Arduino site also does a fantastic job of walking you through the steps of getting your board set up, including the critical steps of getting the board and computer talking in the first place. There are only two main stumbling blocks that pop up: You have to be sure you identify your board correctly; and you have to identify your COM port correctly.

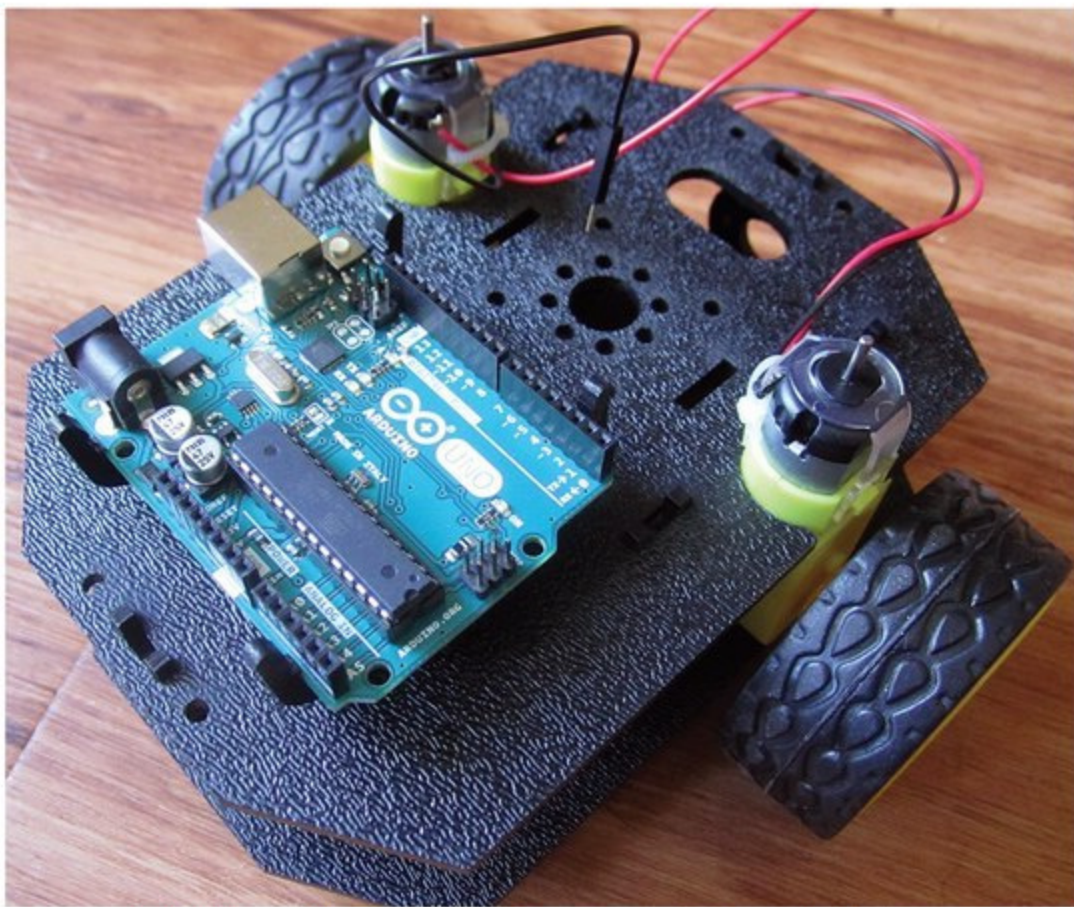
The Duemilanove is one of the older Arduino boards, and you need to be sure to download the required drivers from the FTDI site. You'll know if you have the drivers already by whether or not the proper COM port appears as an option in the relevant drop-down menu.

We started with some programs from the Arduino sample library, including one that allowed us to control the bot with our computer keyboard through the serial monitor. It was fun to see the Bogie spring to life and start tackling

left to Hercule Poirot and his trusty sidekick, Arthur Hastings. We often seem to run afoul of problems getting the bot and the computer to talk to each other, or ensuring that we have all of the required code libraries, or some other unclassifiable issue that stymies our efforts. We have never had those problems, however, while using Arduino boards. The forum



TAKE THE LIME AND RUN.



DON'T FORGET ABOUT THE SPROUT.

terrain with its suspension.

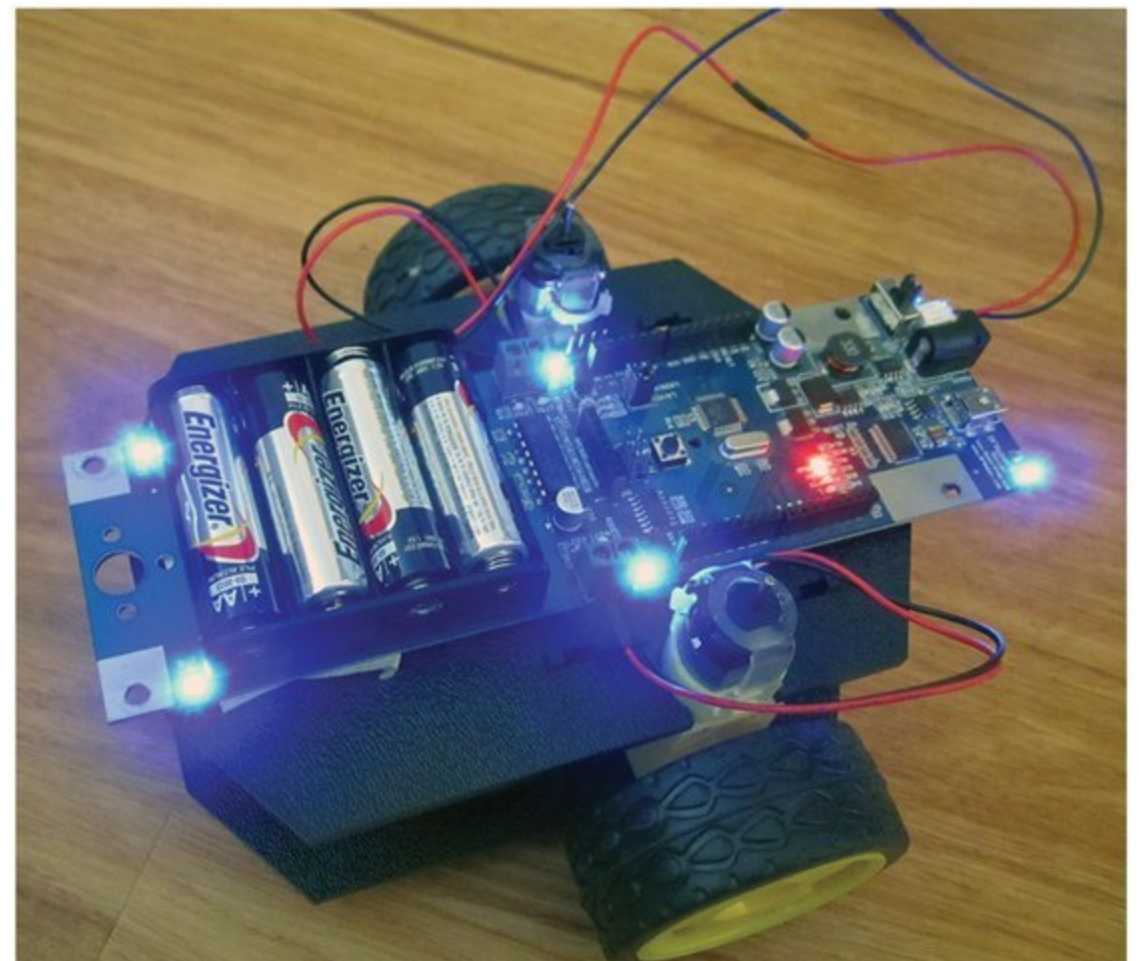
To allow the Bogie to tackle some terrain untethered, we uploaded a simple dead-reckoning program and set up a simple obstacle course. The bot tackled binders and ramps with ease, all while keeping the center platform nice and steady.

To expand the functionality of the Bogie, we added the Actobotics parallel gripper. With the ample mounting holes on the Bogie, it was easy to slap it on the front of the bot, and it only took a few lines of code to bring the claw to life. We were pleased to see that the claw was able to grip its precious payload (in this case, a lime) while it drove over the terrain.

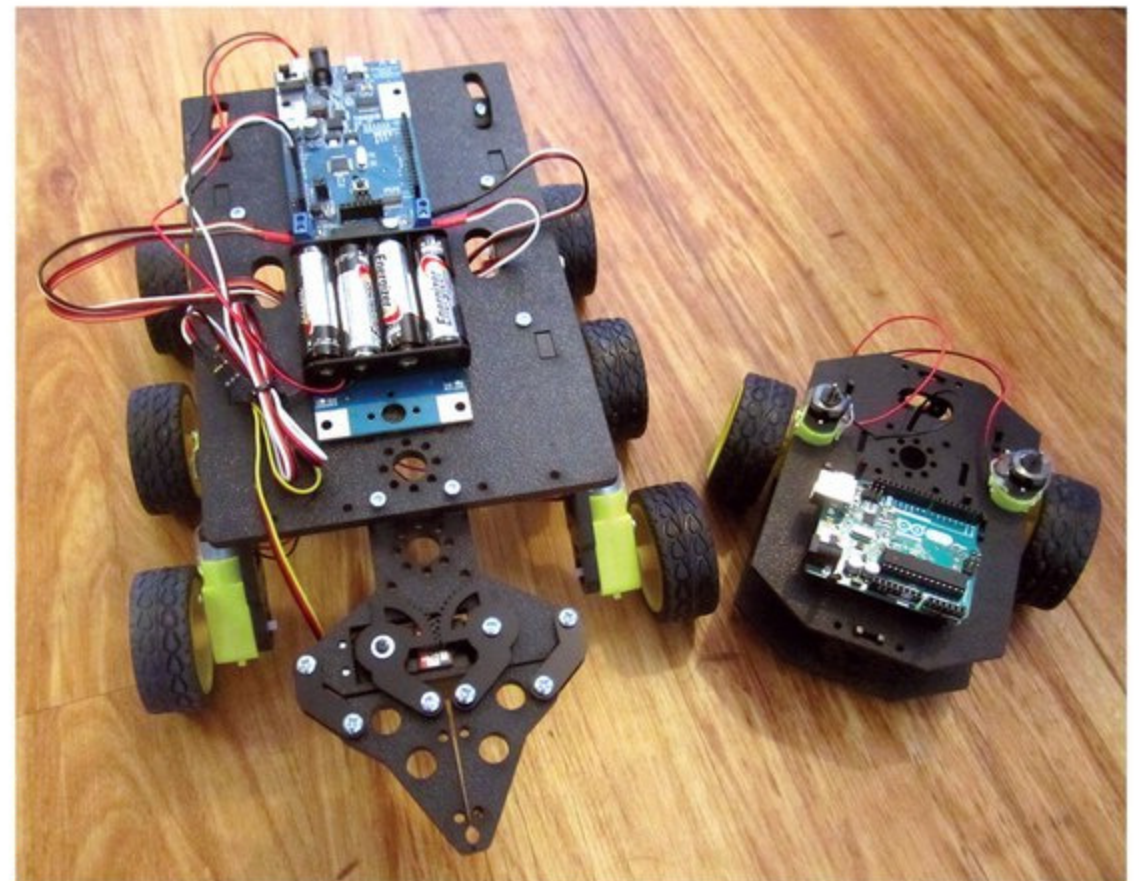
The Power is Yours

The Runt Rovers are meant to be the ideal platform for novice roboticists to get started with robots. Can a kit with DC motors and microcontroller boards be accessible to whippersnappers that are more used to the simple charm of things like LEGOs? We think it can be. As known robotics fanatics, we actually get a lot of questions from friends and coworkers about what might be a good robot kit to get their kid or niece or nephew. A lot of times the requestor is someone looking for something cool and fun that has more substance than just another video game.

We're usually informed that the kid is precocious but that they don't want to hit them with anything too technical or overwhelming. We really think the Runt Rovers fit those criteria to a tee (and also the other very important criteria of not wanting to break the bank on a present). Something like the Sprout goes together with no tools, and as long as you have a board (and a motor driver), you're off to the races. As long as the new roboticist can copy and



MORE BRAIN THAN IT KNOWS WHAT TO DO WITH.



FAMILY PORTRAIT.

paste code, the robot will be up and running. Then, curiosity will inevitably strike — what if I change this character here, or add a line there? All of a sudden, you have a new roboticist excited because they've made a small electronic device come to life all by themselves. We can't think of a better gift than that. **SV**

RECOMMENDED WEBSITES

www.servocity.com/html/robots__accessories.html

Check out the YouTube Video
https://youtu.be/_DARUIObjLE

Police and Security Robots

In this day of media hype about the evils of unmanned aerial vehicles (UAVs) — or drones, as they are now being called by everyone — many people are quite wary of robots amongst them. So much has been written about UAVs that I am going to sidestep these flying robotic vehicles and concentrate on ground-based robots used by police and security forces. Most of these types of robots exhibit amazing capabilities such as the ability to traverse difficult terrain, and they possess powerful manipulators. Many people consider military robots to be in the same category as police robots, but the needs of the military and law enforcement are quite different, as are the requirements for security robots.



Figure 2. *RoboCop* exoskeleton suit.



Figure 1. LA county Sheriff Department's Remotec Andros robot.

Six years ago, I covered (at the time) the state-of-the-art of law enforcement robots, but it is amazing just how fast "state-of-the-art" marches on. Many six year old robots are actually still in use by law enforcement agencies, such as the Remotec Andros F6 series robot

shown in **Figure 1**. This robot company first formed in 1980 and joined with Andros in 1986.

The twin silver barrels are not guns, but 'disrupters' used to disable or to safely explode a suspected explosive device. This robot (used by the Los Angeles, CA Sheriff's Department) has proved very useful in many tenuous situations throughout the years. Improvements in sensors, video feedback, and control systems have given many advanced robots a bit of autonomy not available in 2009.

With the many incidents of police shootings in the news these days, the use of police robots and video recording cameras worn by police officers is becoming prevalent.

Movies haven't really portrayed police robots as effective or even friendly assistants to security forces. The 'robot' portrayed in the 1987

RoboCop movie (and shown in **Figure 2**) was more of a cyborg/exoskeleton than a true electro-mechanical robot.

If you remember the story line, the police department of crime-ridden Detroit of the future attempted to use a huge crime-fighting robot — the ED-209 shown in **Figure 3** — to quell the rampant violence, much to the distaste of the public. When policeman, Alex Murphy was killed, part of his 'resurrected' body and head were used within a steel exoskeleton to become RoboCop. Murphy's robotic alter ego became very successful, much to the angst of the supervillan, Boddicker.

Unfortunately, this and so many

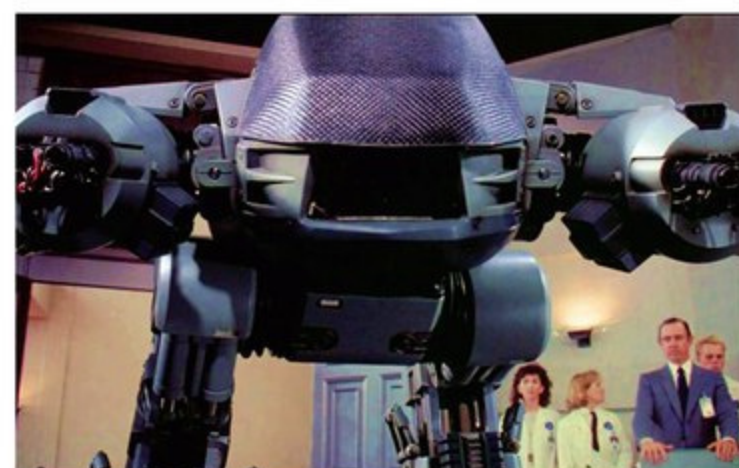


Figure 3. The massive ED-209 police robot from *RoboCop* circa 1987.

Post comments on this article at www.servomagazine.com/index.php/magazine/article/august2015_ThenNow.

movies afterwards portray police robots as uncontrollable killing machines.

Another company — Omni Consumer Products — publicized an interesting police robot concept called the C.R.A.B., shown in **Figure 4**. Stating that the “robot is an all-in-one solution for the safety of future civilization featuring a superb combination of a patrol police officer, armed guard, and a patrol police vehicle without the need of a driver. It will revolutionize future police force(s) with advance(d) robotics and AI.”

The designer, Jamie Martin is said to have derived the design from the Odetics Odex 1 — a very unique robot that I first wrote about 11 years ago. I assume that people still seem to feel that police forces need a very large walking robot to intimidate potential bad guys, and a giant crab-looking thing is even scarier. It definitely looks great for a science fiction movie prop.

What do Robots Offer Security and Law Enforcement Agencies?

Let's look at an officer's daily exposure to deadly force used against *them*. Outside of the military and firefighters, there are few jobs that expose an employee to the possibility that they might lose their life on a daily basis. With that fact in mind, I'll examine a few scenarios of police officers at work.

Last June in Troutdale, Oregon's Reynolds High School, two students were killed: one was the victim, and the other was the shooter. The Troutdale Police Department used a small mobile robot to investigate a restroom where the 'shooter' was last seen. He was found deceased, but the police had no way of knowing this before sending the robot into the restroom to visually determine the situation.

The robot was not armed and was just used to give law enforcement a close-up 'first person view' of the situation without endangering their own lives. Also known as 'remote person view,' this capability is one of the law enforcement robot's most valuable features. Looking at another potential scenario, a person in a typical small American city discovers a backpack lying in a corner of a public building. Smartly deciding to not pick it up or examine it, they notify the local police department.

The police decide to evacuate the building and utilize their new 'robot' to grasp the backpack. Using remote control and video feedback from the robot, they carefully move the backpack to a safe outdoor area. Lacking a true 'disrupter,' they attach a standard shotgun to the robot and remotely trigger the gun aimed at the backpack.

A series of shredded Civics, English, and other textbooks, plus a mangled video game CD fly through the air. A high school student learned a valuable lesson; a police force and a community breathe a sigh of relief.

What do Police and Security Organizations Need in a Robot?

Six years ago, I talked with a few police officers and a security company — including several that used robots in their work — and made a list of some of the basic robot requirements that these groups had back then, but have since modified a few of them.

1. Low cost (that is probably a given for any organization).
2. Ease of operation — minimal training required.
3. Simple to maintain (easy to diagnose problems, locate, and install replaceable components).
4. Light weight with functional handles — able to be maneuvered by one person (or two for larger robots).
5. High resolution color video



Figure 4. C.R.A.B. robot concept by Omni Consumer Products.

system with audio feedback.

6. Two-way audio and video with LCD display onboard the robot for interactive remote communications with a possible suspect.

7. Tether for power and remote control (reel on a robot?).

8. Manipulator arm with gripper to pick up or deliver items, break glass, hold a camera, etc.

9. Ability to easily add additional capabilities such as an explosive disrupter gun.

Additional capabilities might include:

10. Lightweight fiber optic cable for reliable control link in EMI areas.
11. Internal battery power option.
12. Able to ascend and descend stairs.
13. Able to traverse debris and uneven surfaces (search & rescue).
14. Weatherized for rain, snow, and long sun exposure.
15. Ability to right itself if turned over (smaller robots).
16. Long duration operation — 12 or more hours on a charged battery.
17. Swappable battery pack for quick operational turnaround.
18. Total autonomous operation.

Differences Between Police, Security, and Military Robots

One would think that all three types of robots could be used interchangeably, but that is not the



Figure 5. Guardium UGV security robot patrolling an airport.

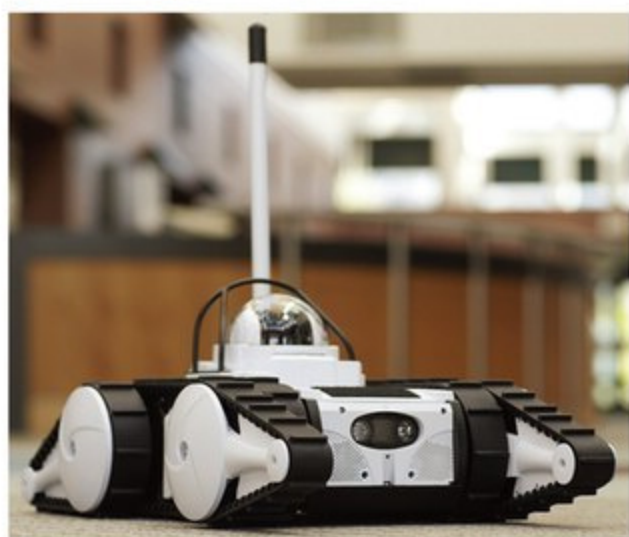


Figure 6. Avatar III by Robotex in Sunnyvale, CA.

case. A typical police robot might stay locked in a mobile trailer for many months with the battery on trickle charge, until it is suddenly needed for a particular scenario. The police robot must be constructed in a way to withstand rough terrain, rough treatment by a suspect, and possibly have some sort of manipulator to handle a suspected bomb or unknown object. It is usually remotely operated by a trained person, and should possess a superior high definition vision system. A security robot is quite often used on a daily or nightly basis, and needs to be continually operable. The robot can look like the IAI Guardium shown in **Figure 5** — a golf cart sized autonomous vehicle patrolling a typical airport.

It can also look like the small Avatar III security robot shown in **Figure 6** that weighs 25 pounds and is made by Robotex.

It must contain high density/long life batteries as it cannot be tethered to a power or control source. It is rarely remotely operated and its sphere of operation is within a known environment. A significant degree of autonomy is important such as



Figure 7. Denning Mobile Robot's security robot.

following a line on the floor or a pre-programmed path. The robot's working terrain is usually within a building or within an enclosed outdoor yard with a fence, and simple steerable tires or differential steering is sufficient in most cases. It would rarely be 'attacked' by anyone and damages most likely would be minimal. The robot might be used to continually send back a video signal of its surroundings to a manned central control station.

It should also be noted that security and police robots need not take a passive 'look and see' role in their operations. I am not implying using weapons. A security robot might use an ear-splitting siren, a brilliant xenon strobe, or even emit a cloud of tear gas in the presence of an unwanted intruder. A police robot might use the same techniques and maybe even a flash/bang stun grenade to disorientate an intruder, but these devices have also been known to start a fire, so use is not always appropriate.

In contrast, a military robot must be designed for both of the above roles, with the possibility of damage (or destruction) most likely. I personally examined an iRobot PackBot that had been returned from the Middle East military conflicts and had an unfortunate encounter with an IED. What I saw was a mangled pile

of robot scrap that was strapped atop a pallet. These robots can take a lot of damage and still come up 'ticking.'

A military robot could be continually operated throughout a theater of battle and might be autonomous or remotely operated. Modern military operations are frequently within city environments with many buildings and structures.

Early Security Robot

Back in the early '80s when I was attending the annual robot shows presented by Robotics International of the Society of Manufacturing Engineers (RI-SME), I first saw a very unique robot designed for security applications. The Denning Sentry shown in **Figure 7** was manufactured by Denning Mobile Robots of Woburn, MA and was very well designed and built.

Standing about four feet tall and 28 inches in diameter, it was a fairly large and heavy robot. The omni-directional wheels allowed the robot to revolve on its axis, or immediately move in any direction (without the body of the robot revolving) and then head in a new direction. **Figure 8** (a photo of the Denning Branch research robot platform) shows how the three wheel assemblies were steered by the motor in the forefront of the photograph. **Figure 9** shows a view of the interior. The robot was preprogrammed for a specific path within a building's structured environment. Navigation was accomplished by sensing strategically placed IR beacons along an intended path to guide the robot in conjunction with the programming. It had a series of 24 Polaroid electrostatic ultrasonic range finder transducers around the robot's circumference to detect stationary objects and walls, as well as moving objects of interest in a security situation.

According to their literature, "These range finders improved the navigational solution by allowing the robot to measure its offset as it approached a wall structure, while

simultaneously supporting a fairly robust obstacle detection capability. More importantly, the ranging sensors facilitated later development of a powerful wall-following algorithm that reduced dependence on the optical beacon system, increasing flexibility in the event of required path changes and reducing the need to modify the robot's environment."

Their show booths always seemed to draw large crowds, and robot engineers were always amazed at the quality and mobile abilities of the Denning robot, as well as the sonar and navigation capabilities. The company supposedly sold a number of robots to several prisons for inmate and premises security. They even made a non-security version called the 'Model T' that was basically a stripped-down security variant used for research purposes. It had several Z-80 processors and a Motorola 68000 as the main controlling processor.

Unfortunately, it was one of those companies with a good product that had development and marketing issues that prevented it from being appreciated by potential customers. In the early '90s, disappointing sales led to the company going into bankruptcy in 1993 and — after a sale to another group — finally disappearing in 1997.

Bart Everett's Robart III

Robart III (**Figure 10**) is one of a series of laboratory prototype mobile robots designed by Bart Everett, Technical Director for Robotics at the Navy's SPAWAR Systems Center in San Diego, CA. Everett has one of the most enviable jobs that I know of. I had the opportunity to spend a couple of days with him on two different occasions at SPAWAR. He showed me many of the Navy's robots, including the mangled iRobot Packbot that I mentioned earlier. Security for our military is extremely important and test platforms such as Robart III are used to perfect surveillance and security, as well as having potential weaponry



Figure 8.
Denning
Branch
MRV2
research
robot
platform.



Figure 9.
Interior
of
Denning
sentry
robot.



Figure 10. Robart III by Bart Everett
of the Navy's SPAWAR Systems.
Photo courtesy of Dave Bullock.

mounted on a mobile robot.

The Gatling gun shown in **Figure 11** is strictly a modified air-soft type of weapon that can shoot 3/16" plastic pellets or darts. The six barrels allow six darts to be fired within 1.5 seconds by the 150 psi air from the white compressed air cylinder shown.

A later version utilized a laser designator for sighting a target. It was not — nor ever was — developed as a production security platform or weaponized robot, but rather as a test bed. Utilizing a Sick LIDAR (in blue) and a series of distance measuring Polaroid ultrasonic transducers — much like the Denning robot — it could find its way among many obstacles on level ground. The Navy currently has in development (and in service) many robots that are far more advanced than this test vehicle.

Newer Robots for Police and Security Applications

Technology has advanced quite a

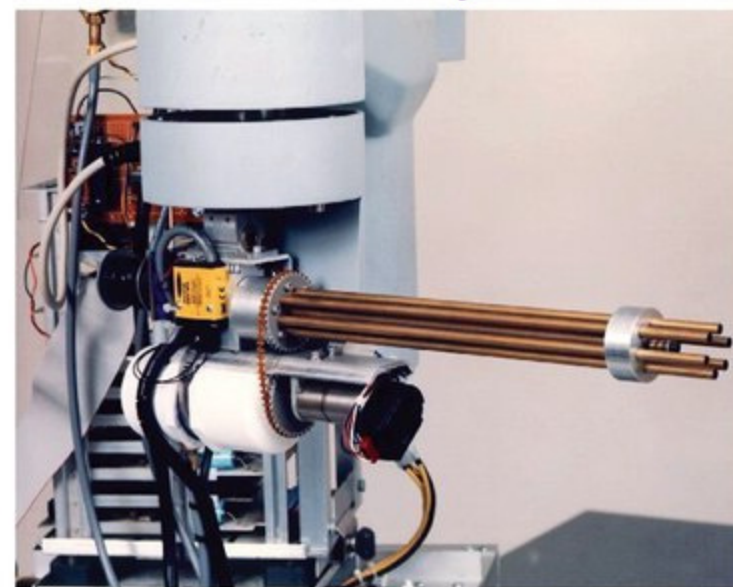


Figure 11. Air-powered Gatling gun on
Robart III.

bit since the early '80s — even in the past decade. Video cameras have shrunk in size and cost from the old Vidicon tube cameras to the small HD CCD cameras that have far better resolution. Adding pan and tilt, along with zoom and low light, and IR capabilities to the cameras have enhanced remote presence/detection for critical situations.

RF control and video feedback, as well as wired and fiber optic cable control has allowed operators to enjoy intricate mastery of the robot's base as well as a manipulator arm and gripper. Off-the-shelf sensors such as accelerometers, gyros, and compasses



Figure 12. Knightscope K5 security robots.



Figure 13. Alsok's Reborg-Q.

have allowed more accurate navigation. LIDAR imaging and ranging sensors have become affordable to fine-tune the robot's navigation suite. GPS and SLAM navigation for accurate 3D mapping along with AI have allowed more autonomy for high-end robots.

Knightscope K5

One of the newest robots to make the security scene is the Knightscope K5. Rather than a robot used to monitor a single business or even a prison, the K5 seems to be dedicated to roaming wide areas to keep watch on people, possibly locating individuals who might be of a danger to the general population.

Looking like *Star Wars'* R2D2 mated with a large egg on wheels, the robot can be seen in **Figure 12**.

The company was formed after the events of the December 2102 Sandy Hook school massacre and the bombings at the Boston Marathon. The company's Chairman and CEO, William Santana Li founded the

company " ... with the ultimate goal of providing an avenue for all Americans to join the fight against crime."

The buzz on the Internet and within different robotics interest groups sees the robot as a mobile video security camera platform — much the same as the many fixed video surveillance platforms that are all around us these days. They see the robot as a way of "outsourcing security." "In order to accomplish that," Li says, "we will provide the public with open, transparent access to real time data and alerts, and allow them to contribute important information to authorities. Together, we will more rapidly and effectively secure our communities and save money while doing so."

Based in Mountain View, CA (in the midst of Silicon Valley), Knightscope is looking at the poor performance of many security employees and the very high 400% turnover rate of these people. They intend to rent the robots at \$6.25 per hour and operate them in continuous 24 hour 'deployments' for periods of one to three years.

The K5 can operate within a structured environment that has many people traveling at no more than 3 mph, whereas a K10 model will operate in a more open area.

It will be an autonomous machine using many of the sensors that I mentioned earlier (such as LIDAR), and possess pattern recognition for license plates and human faces. It also has directional microphones for additional detection methods. Wheel odometry, inertial measurement units, GPS, and basic proximity sensors will assist in robot navigation, as well as human and obstacle avoidance.

This real time data will be stored and used by law enforcement agencies. It is not intended to replace human guards or police officers, but rather to enhance their capabilities.

There are no weapons attached to the robot and it was designed to be a bit harmless in appearance. Knightscope calls the K5 an

autonomous data machine. Time will only tell if this type of security robot will incur the wrath of the general population in the same manner as the drone — especially with the K5's many intrusive data-gathering capabilities.

ALSOK Reborg-Q

Another interesting security robot from Tokyo — the Reborg-Q shown in **Figure 13** — has been in the news since 2006. Alsok began research on security robots back in 1982 with the intent of increasing the efficiency of manned security. The Reborg is an updated version of their earlier Guardrobo, and weighs about 200 pounds. It can be programmed to follow a patrol course or can be remotely controlled via an RF link and a joystick controller. It is able to push the required buttons to use an elevator. There are four cameras in the head and shoulders to view, record, and send back images and video to a manned security center. Human presence (ultrasonic and PIR) sensors can detect people, water leakage, and fire. At that point, security personnel can review the images and make an appropriate response.

A touchscreen is mounted in the robot's chest to display weather updates, lost child information, community danger messages, and even promotional messages. A card reader mounted on the robot can permit it to serve as a guard at a gatepost to allow or deny employees or others entry into a business or other facility.

Final Thoughts

The unrest in today's world has made advanced security and police robots a must to protect our communities and society. I have concentrated on security robots and concepts, but interested readers should delve deeper on the Internet to discover some amazing robots that will soon be prowling our streets and buildings to protect us. **SV**

FEED
STIMULATE
SATISFY
YOUR MIND

APPETIZER

Battling Robots Isn't It Violent? No, It Isn't.

by Erica Smentowski

As a veteran BattleBots builder's wife and former BattleBots employee and a mother of two boys, I'm thinking that some mommies out there hesitate to jump on the fighting robots bandwagon because they think "it's violent, and we don't allow violence." When I started posting news about the new BattleBots show, I saw a lack of Facebook "likes" from a certain demographic. The mom demographic.

So, let me say to them what we've been saying for, oh, about 18 years. Battling robots are not violent, but they are destructive. Why promote that?

Well, what did your toddler do with every block tower? They **KNOCKED IT DOWN**. Blocks everywhere. And then what did they do? They **BUILT IT AGAIN**. AND AGAIN. Even if the blocks were all over the room.

Now, let's talk about culture today. Everything is disposable. You don't fix it; you throw it away. Appliances, relationships, you name it. Why bother to salvage anything?

This sport — and we do consider it a sport — involves people using all of their tinkering, engineering, and imaginative skills to apply math and science (or maybe just a good working theory) to bring a machine to life.

After working on it for hundreds of hours, they know it is going to get destroyed in some way, shape, or form. They know it is not going on the shelf to be displayed and preserved.

So, they bring their creation into the arena where something destructive inevitably happens to it. It gets its armor shredded, frame bent, maybe expensive electronics go up in smoke and flames, etc., in an attempt to beat the opponent. You win some, you lose some.

Do you throw away your robot afterwards? No. You rebuild it. Sometimes with only a short timeframe before the next match, you need to improve your bot and quickly learn from your mistakes.

You look for design flaws and ways to beat your next or most formidable opponent. Sometimes your bot literally limps back into the arena to advance to the next fight, and you head back into battle.

It can be nerve wracking. I often find myself shaking, a lump in my throat, adrenaline off the chart when I see my husband's robot in the BattleBox before the start of a fight.

But is it violent? When I worked in the BattleBots office, I fielded fan emails and phone calls. I will always remember the communications I got from parents saying, "My kids were watching wrestling and jumping off the couch trying to kill each other. Now, I can't get them away from their LEGOS, because instead, they're designing fighting robots! Thank you!"

No, it's not violent. And schools have caught on. Robotic combat programs are alive and well in high schools. Our current online store

Post comments on this article at www.servomagazine.com/index.php/magazine/article/august2015_Appetizer.



(www.robotmarketplace.com) is a supplier for these schools, so I know firsthand. I personally believe that the combat element is more impactful than say, building a robot to complete a task.

There is deep, intense emotion involved in seeing your creation get broken or mangled, and that helps you deal with the inevitable tragedies in life differently than say, the disappointment in scoring low points on a task.

You have to sort out all that raw emotion and salvage what you can of your robot from sometimes literally pieces of scrap. Hopefully, you come back stronger.

That's not violent, that's a life skill. It holds incredible life lessons. So, let your kids watch the battling robots because it will only spur them on towards imagination, engineering, science, and hands-on tinkering.

If they knock down the block tower, they will always want to rebuild it. **SV**

- Bot wife and widow since Robot Wars 1997
- Former BattleBots employee from Comedy Central era
- Currently co-owner of www.robotmarketplace.com with husband, Jim Smentowski of Team Nightmare (BattleBots)

ROBO-LINKS



MaxBotix
High Performance Ultrasonic Rangefinders

www.maxbotix.com
info@maxbotix.com Phone 218-454-0766

RobotShop
.com



superbrightleds.com
COMPONENT LEDs • LED BULBS • LED ACCENT LIGHTS



AndyMark
Inspiring Mobility



www.andymark.com

For the finest in robots,
parts, and services, go to
www.servomagazine.com
and click on **Robo-Links**.

ramsey
www.ramseykits.com
AM/FM Broadcasters • Hobby Kits
Learning Kits Test Equipment
...AND LOTS OF NEAT STUFF!

GET THE NUTS AND VOLTS DISCOUNT!
Mention or enter coupon code **NVRMZ142**
and receive 10% off your order!



INVEST in your BOT!



HiTEC
12115 Paine Street • Poway, CA 92064 • 858-748-6948 • www.hitecusa.com

Need electronics?
www.nutsvolts.com

**Did You Know Preferred
Subscribers get access to all the
digital back issues of
SERVO Magazine for free?**

Call for details
1-877-525-2539

ADVERTISER INDEX

All Electronics Corp.	11	Parallax	15
AndyMark	10, 82	Pico Technology	Back Cover
Firgelli	22	Pololu	3
Front Panel Express LLC	57	RobotShop, Inc	82, 83
HiTec	2, 82	Robot Power	15
M.E. Labs	35	SuperBrightLEDs	82
Maxbotix	82	The Robot Marketplace	9
PanaVise	44	Tormach	26



Imagine it. Build it. Control it.™ Fly it!

Introducing the Quadrino Nano UAV Flight Controller



Coupon
\$10 Off*
QFC2015

- MS5611 Barometer with foam cover
- Venus 838FLPx 50Hz GPS

*Limited time offer.

Available at www.Lynxmotion.com and www.RobotShop.com



Lynxmotion is a trademark of RobotShop.

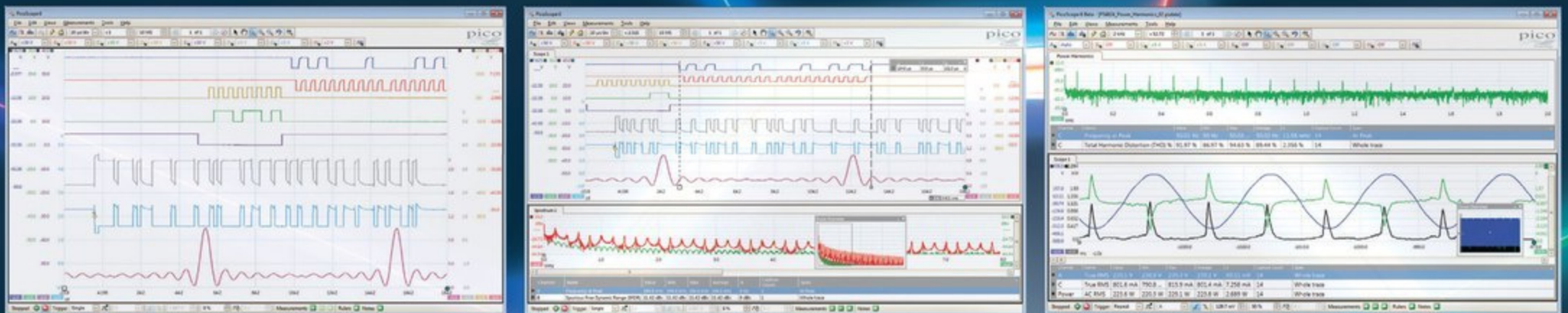




7 functions of PicoScope:

1. Oscilloscope
2. Spectrum analyzer
3. Function generator
4. AWG
5. Logic analyzer
6. Serial protocol analyzer
7. Automatic waveform test

8 CHANNEL PC OSCILLOSCOPE



For just \$2300

- High resolution
- USB powered
- Deep memory



INCLUDES AUTOMATIC MEASUREMENTS, SPECTRUM ANALYZER, SDK, ADVANCED TRIGGERS, COLOR PERSISTENCE, SERIAL DECODING (CAN, LIN, RS232, I²C, I²S, FLEXRAY, SPI), MASKS, MATH CHANNELS, ALL AS STANDARD, WITH FREE UPDATES

12 bit • 20 MHz • 80 MS/s • 256 MS buffer • 14 bit AWG

Call Us At: 1-800-591-2796 www.picotech.com/pco541