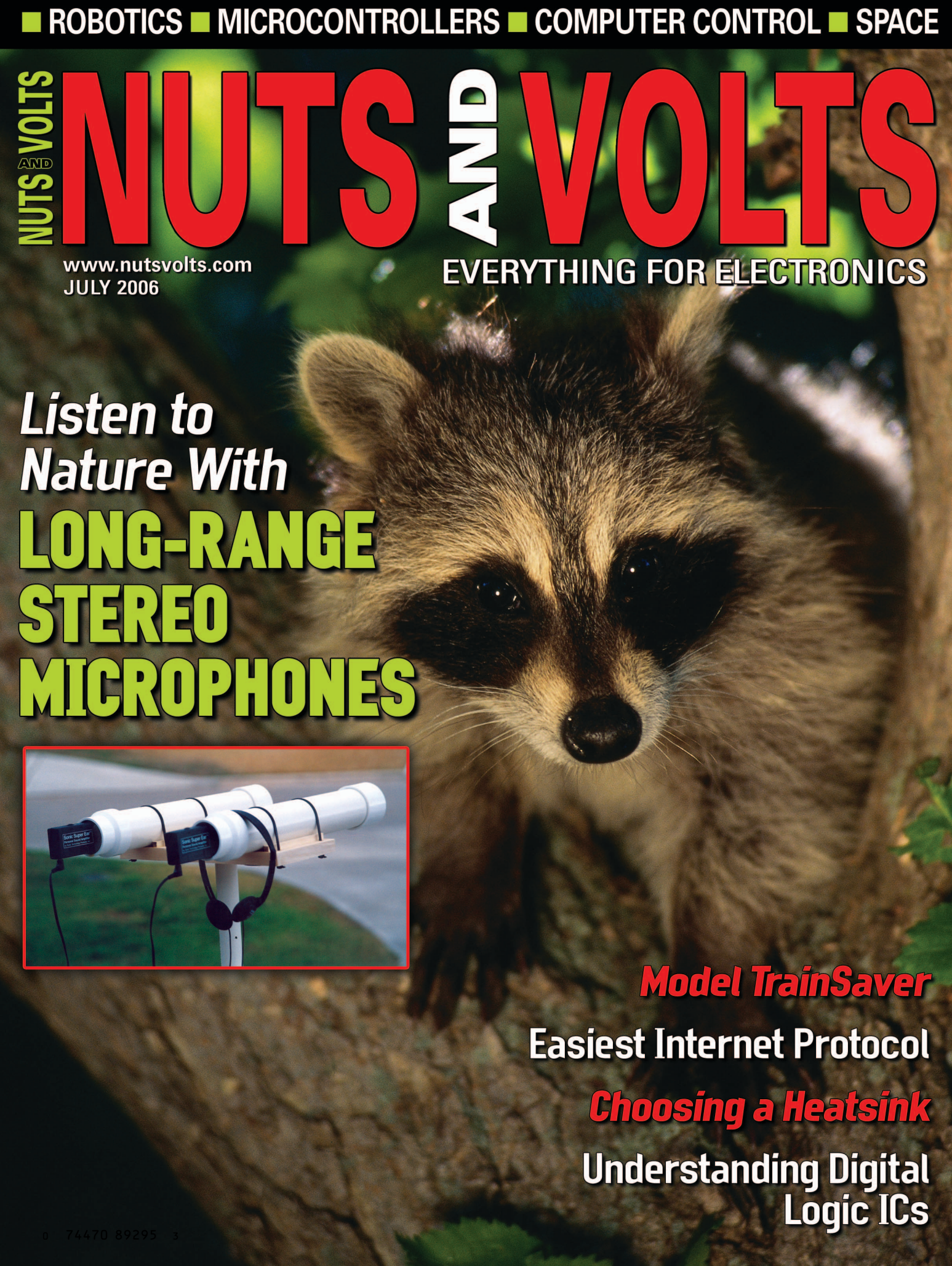
■ ROBOTICS ■ MICROCONTROLLERS ■ COMPUTER CONTROL ■ SPACE

NUTS AND VOLTS

NUTS AND VOLTS
www.nutsvolts.com
JULY 2006

EVERYTHING FOR ELECTRONICS



Listen to Nature With

LONG-RANGE STEREO MICROPHONES

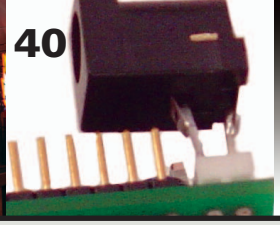
Model TrainSaver

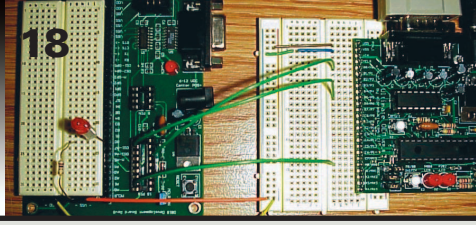
Easiest Internet Protocol

Choosing a Heatsink

Understanding Digital Logic ICs

0 74470 89295 3





NUTS AND VOLTS

JULY 2006

PROJECTS and FEATURES

36 LONG-RANGE STEREO MICROPHONES
Listen to faraway sounds using this easily built microphone/amplifier assembly.
■ *By J. Ronald Eyton*

40 CONTROL YOUR WORLD
Interface your PC to the CMUcam.
■ *By Michael Simpson*

46 THE TRAINSAVER DIGITAL ELECTRONIC CONTROLLER
This device will reduce wear and tear on model train locomotives.
■ *By Vern Graner*

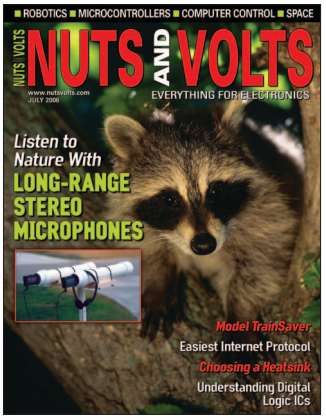
59 RADAR AND ELECTRONIC WARFARE FUNDAMENTALS
Learn basic concepts of radar and see how EW develops from that.
■ *By Gerard Fonte*

67 CHOOSING A HEATSINK
Learn the basic process to keep the right components cool.
■ *By H. Ward Silver*

74 UNDERSTANDING DIGITAL LOGIC ICs
Part 1: Digital Logic IC Basics.
■ *By Ray Marston*

ON THE COVER ...

This easy-to-build microphone setup can bring more than just our furry and/or feathered friends close up and personal.



COLUMNS

08 TECHKNOWLEDGEY 2006
Events, advances, and news from the electronics world.

12 Q&A
Serious questions are answered, a very cool NASA website, and more phone busy stuff.

18 GETTING STARTED WITH PICs
Developing with a PIC bootloader.

26 PERSONAL ROBOTICS
Design a Mini Sumo — Part 1.

82 STAMP APPLICATIONS
A Tale of Two Props.

86 NEAR SPACE
The Geiger counter telescope — Part 2.

90 THE DESIGN CYCLE
The easiest Internet protocol of all.

DEPARTMENTS

06 READER FEEDBACK

32 NEW PRODUCTS

56 SHOWCASE

64 NEWS BYTES

66 ELECTRO-NET

96 CLASSIFIEDS

100 NV BOOKSTORE

102 TECH FORUM

105 ADVERTISERS INDEX

Nuts & Volts (ISSN 1528-9885/CDN Pub Agree#40702530) is published monthly for $24.95 per year by T & L Publications, Inc., 430 Princeland Court, Corona, CA 92879. PERIODICALS POSTAGE PAID AT CORONA, CA AND AT ADDITIONAL MAILING OFFICES. POSTMASTER: Send address changes to **Nuts & Volts, P.O. Box 15277, North Hollywood, CA 91615** or Station A, P.O. Box 54, Windsor ON N9A 6J5; cpcreturns@nutsvolts.com

The TrainSaver is a device that reduces wear and tear on model train locomotives used in commercial environments.

The TrainSaver maximizes the life of the engine by only running the train on timed intervals, and only when people are present to view the train's performance. This reduces the cost of ownership of the train system by increasing the life of the engine. The device also adds synchronized sound effects to enhance the train's visibility to the audience.

THE TRAINSAVER DIGITAL ELECTRONIC TRAIN CONTROLLER

It All Started With a Hankerin' for Bar-B-Que ...

The PokeJo's BBQ restaurant that my fellow workers and I frequent here in North Austin, TX, has a nifty electric train that circumnavigates the entire restaurant. As we entered the establishment one afternoon, we noticed the train was missing! A short chat with the manager revealed the train would break down from continuous use and had to be sent to a repair shop. Moreover, the cost of repairing the train engines had gotten high enough that the management had canceled plans to add trains to any of their other locations and was even considering retiring the existing train from our favorite restaurant!

I began to mull over the problem and decided I should be able to rescue the train from abandonment while maybe at the same time enhancing its value to the restaurant. A few inquiries with a local hobby shop showed there might also be other applications for such a solution in enhancing the life of other toy trains used in commercial environments. It seems that the motor and gear train in many consumer grade trains are not designed for the continuous use to which they are put when placed in a commercial venue. Since there are a limited number of hours their gearboxes and motors could provide, we needed to find a way to get the most out of them.

Is it Train Time Yet?

Initially, my design revolved around various timer circuits to reduce the total amount of time the train spent in a powered condition (presumption being less run time = longer life). I built a simple 555 timer circuit to start and stop

Acknowledgements

■ I would like to thank the following people who were critical in making the TrainSaver a reality: Jim Huggans, Sharon Sudduth, Nic Graner, Paul Atkinson, Denise Scioli, The Robot Group of Austin, and the good folks at the PokeJo's BBQ restaurant.

Prop-1™, BASIC Stamp® are registered trademarks of Parallax, Inc.

the train on intervals, but on testing, I discovered this approach was less than adequate. With no positioning feedback, the train would stop at random points around the track when power was removed by the timer. It was also not very realistic as it seemed as if the train had malfunctioned when it would stop or start for no apparent reason in mid-lap around the track.

Some patrons actually approached the cashier to point out that the train had broken down. Worse yet, my simple timer approach would not allow the staff to start the train on demand for folks who would ask to see it run (as a treat for someone's birthday, for example).

Having the train run and stop in a predictable manner, combined with the ability to start the train on command, would seem to need something more sophisticated than a simple timer. In fact, it seemed to call for a microcontroller. Coincidentally, it would also give me an excellent opportunity to create a project around the new Prop-1 I had purchased (see Figure 1) from ParallaxEFX — a new division of Parallax that specializes in controllers for animatronics and entertainment devices. The Prop-1 microcontroller seemed to be a good match for this project for the following reasons:

- It is low cost at only $35 for the complete, assembled unit.

- It has an on-board ULN2803a high-current Darlington chip capable of driving inductive loads.

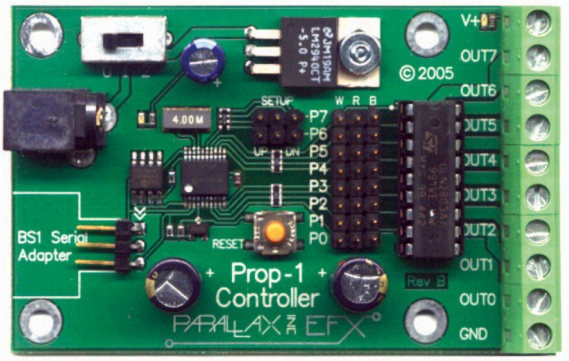
- The ULN2803a is socketed so this critical part would be field replaceable.

- With eight I/O pins, it has plenty of device control available.

- Using PBASIC, it's easy to program.

Of course, once we have a microcontroller at our command, we have a slew of new capabilities that can add fun and value to the project. For example, we could utilize Pulse Width Modulation (PWM) to gently start/stop the motor. This should enhance the life of the motor and the gear train since it would reduce the instant torque exerted on the gear box when power was applied/removed. It would also make it trivial to allow direct control of the train so the restaurant staff could trigger it to perform on command.

■ FIGURE 1. The Prop-1 from ParallaxEFX.



With a microcontroller, I could add a position sensor to have predictable stop and start points and also count the laps the train had run. At first, I imagined using Hall-effect sensors or some other physical method to sense the train's position, but it seemed simpler and possibly more reliable to use a touch free sensor. I decided that an IR LED/IR phototransistor arranged in a beam-break style would provide a good way to determine the train position without requiring any modifications to the train or the track.

Also, most train track setups have a station, trestle, tunnel, or some other feature where the IR LED and sensor could be mounted, so this seemed the best approach. In our case, there was an obvious section of the track where the train should stop, so the optimal position for the sensor was easy to determine (there is a tunnel that would hide the train from view).

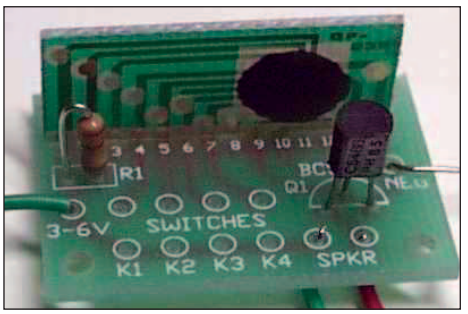
Since the train would be integral to the ambiance of the establishment, we could have a self-start mode that would cause the train to run one or more laps if it was left idle for a software configurable period of time (i.e., not manually triggered by an employee in a 10-minute period). Since the idea was to have the train operate without attention from the staff, the user interface should be as simple as possible consisting entirely of a single button and a single indicator LED. So I had my list of requirements ... or did I?

What's a Train Without a Whistle?

I tried out my prototype design at the restaurant, and I was able to have the train run on command, run on unattended intervals, and stop and start from a consistent location. I even had a small PWM routine in the code that would ramp the power up on start and down on stop to ease the strain on the motor/drive train. However, as I observed my prototype design in action, I noticed that in many cases people would not notice the train had started, or were puzzled when the train would just disappear without apparent reason.

It occurred to me that having some sort of sound effect would add to the realism of the train and bring people's attention to it when it was

■ FIGURE 2. The Chip On Board (COB) Train sound effects sound module fully assembled.



■ FIGURE 3. Train Sound Effects Kit components.

about to perform. For variety, I could use specific sounds to indicate when the train was stopping or just passing through the station. Since the Prop-1 had some unused outputs in my existing design, I decided to look for a sound module.

Sound, Sound, Everywhere the Sound

A quick web search turned up a very inexpensive COB (Chip On a Board) digital sound generator (see Figure 2). This little sound module kit is widely available from various manufacturers online. The Train Sound Module Kit comes with the sound generator itself, four NO (normally open) pushbutton switches, a small motherboard on which the chip is mounted, and a 3", eight-ohm speaker (see Figure 3).

This neat little module plays back digital samples of various train sound effects including a crossing bell, the clickety-clack of the train wheels on a trestle, and a steam train whistle. Due to the nature of the switch inputs, it would be trivial to trigger the sounds using pins on the Prop-1. Since I can establish the position of the train using the IR beam-break sensor, I would be able to trigger the various sound effects to occur when the train was near the speaker, thus enhancing the illusion the sounds were emanating from the train itself.

■ FIGURE 4. Eight watt audio amplifier kit.

Whistle With Laryngitis?

After I assembled the COB sound module, I noticed it came with a rudimentary audio amp in the form of a single transistor output. On my bench the sound was audible, but certainly didn't play with much authority. When I put the prototype in place at the restaurant, you couldn't hear the module at all! The single output transistor on the module just wasn't putting out enough power.

I decided to beef it up by ordering an eight watt audio amplifier kit (see Figure 4). This little circuit puts out up to eight watts of power into eight ohms — a good match for the speaker included with the sound module kit. It gave me the punch I needed so the train sounds could be heard throughout the restaurant.

Is There Anybody ... Out There?

The train was now able to be called into action on demand, play sounds that would relate to its actions, and run automatically on timed intervals, I thought I was finished adding features. However, one of the folks working at the restaurant pointed out that the train continued to run even if the dining area was empty of patrons. If we could cause the train to sleep when no one was in the dinning hall area, we could further extend the life of the engine and also reduce the annoyance factor for the staff (they do get tired of hearing the bells and whistles of the train after a while).

I added a motion sensor to the Prop-1 and added some code to keep track of the amount of time elapsed since the last time the motion sensor had seen movement. If that time amount reached a predetermined minute threshold, the train would be put to sleep and would cease to auto-add laps. This also made it possible for the train to sleep overnight in the event the staff forgot to turn the train off at the end of the day.

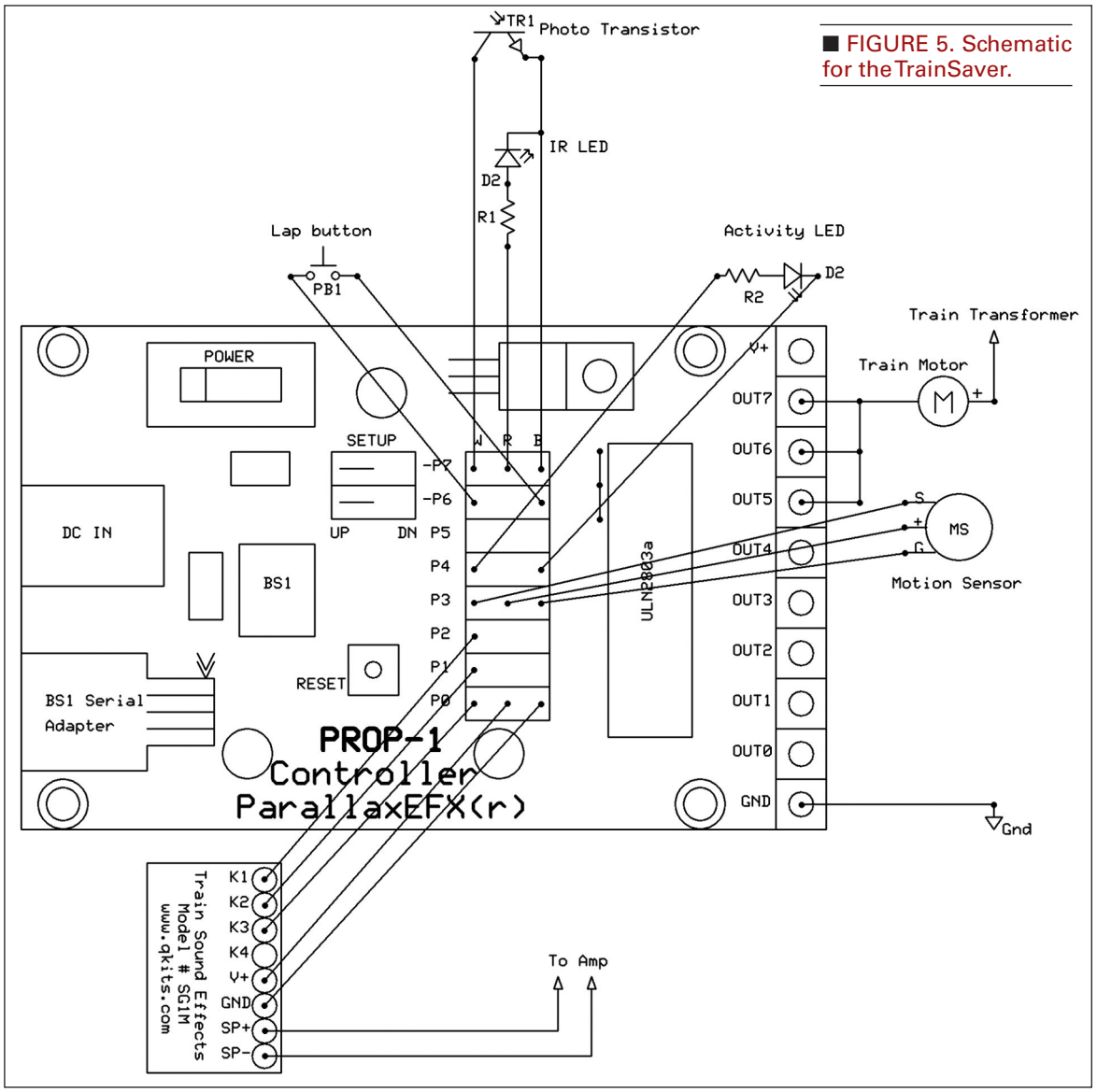
Put Your Parts Together Now!

So, now that I had all my parts and some successful testing under my belt, it was time to break out the soldering iron and create the final version that would be permanently mounted at the restaurant. The Prop-1 comes pre-assembled with the ULN2803a, the LM2940 +5V regulator, and .100 spacing headers for each of the I/O pins. It would be the heart of this project. I allocated the Prop-1 pins in this way (see Figure 5):

- Pin 0 — Sound Module (Crossing Bell sound)
- Pin 1 — Sound Module (Clickety/Clack sound)
- Pin 2 — Sound Module (Whistle sound)
- Pin 3 — Motion Detector
- Pin 4 — LED Indicator Light
- Pin 5 — Power to Train Motor
- Pin 6 — IR Phototransister
- Pin 7 — NO Pushbutton

The assembly of the components into a small finished box is fairly straightforward. I started by soldering together the COB sound module kit, then the eight watt amplifier kit. With the two kits assembled, I now needed to fabricate the IR transmitter and receiver.

The pin spacing on the Prop-1 is .100" and is a perfect match for extra



■ FIGURE 5. Schematic for the TrainSaver.

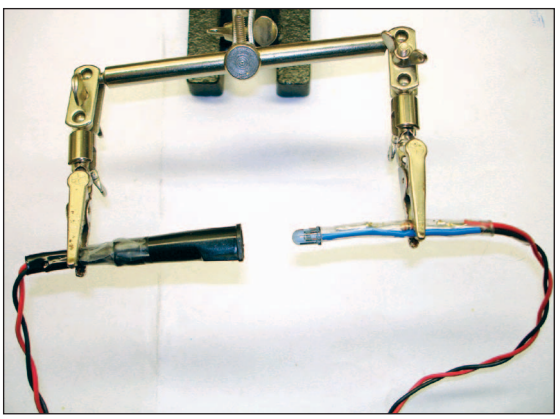
CD-AUDIO cables I happen to have laying around, so I decided to salvage them for their connectors. Since the Prop-1 has +5V (labeled "R"), Gnd (labeled "B"), and IO (labeled "W") arranged for servo connection, I would have an easy source for power and ground for the sensor and for the source IR LED.

I found some three-conductor wire from an old mic cable in my junk box and added the .100 spacing header to one end. I then created two pig-tails for the IR LED and the IR phototransistor. I used an old pen cap as the ambient light shield for the phototransistor. I placed a current limiting resistor in-line near the LED and then wrapped it all up nicely with heat shrink tubing (see Figure 6).

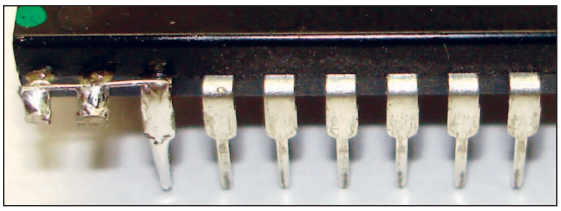
Now, with the sensor squared away, I needed to mount all the parts in the case. However, during the preliminary testing I noticed the ULN2803a was getting rather hot when switching the train. Though ULN2803a is a versatile and robust part, when I measured the current drawn by the train, it proved to be a bit in excess of the recommended max current draw for a single pin on the ULN2803a.

To make sure we had enough headroom and wouldn't burn up the chip, I ganged multiple pins of the ULN2803a to get higher current capability. The fact the part is socketed made this process easy since all I had to do was remove the ULN2803a, bend pins 1 and 2 out to 90-degree angles (see Figure 7), bridge pins 1, 2, and 3 with a bit of solder, and then replace the chip.

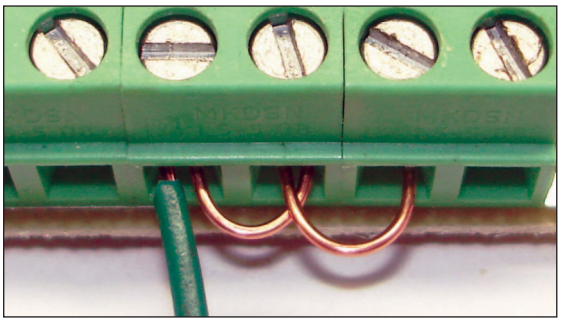
I then bridged the three corresponding ULN2803a output screw-terminals (outputs 7, 6, and 5) on the screw terminals of the Prop-1 (see Figure 8). This modification allowed the BS1 to send the PWM output to three input pins on the ULN2803a using only pin



■ FIGURE 6. IR beam-break detector prototype. A pen cap is used to reduce ambient light sources.

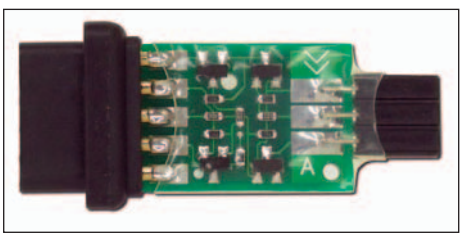


■ FIGURE 7. The ULN2803a pins bridged to allow a Prop-1 output to control three of the ULN2803a outputs.



■ FIGURE 8. Prop-1 output terminals bridged to provide higher amperage output.

ULN2803a's ability to handle current is based on total part power dissipation. If your train motor will draw more current that the chip can source, it will destroy the chip. I recommend you examine the ULN2803a data sheet to determine if your intended current draw will exceed the part specifications. If your train draws more current than the chip can handle, you might want to consider using the Prop-1 outputs to drive a relay or a high-power MOSFET instead.



■ FIGURE 9. Parallax serial adapter. Using this, you can program the Prop-1 using a standard serial output from the PC.

5, but ganging outputs P5, P6, and P7 for a combined load capacity of approximately 1.5 amps.

NOTE: Please be aware that the

■ FIGURE 10. The Prototype Prop-1 open for inspection. Note that the LED indicator is integral to the pushbutton switch. I used this combo-part since I already had it in my junk bin. You can use a discrete LED and one of the pushbuttons that comes with the Train Sound Kit.

My newly modified Prop-1 board was now ready to be mounted in the case. I used a dab of hot-melt glue to hold the Prop-1 to the bottom of the case and then cut a hole in the side of the cabinet big enough to accommodate the BS-1 serial programmer (see Figure 9). I connected the sound module's power to the Prop-1 P0 "R" and "B" power connectors. I then connected the sound module's K1, K2, and K3 pins to the Prop-1 P0, P1, and P2 "W" connectors (see Figure 10). Now that I had the connections made, I used another dab of hot-melt glue to affix the sound module to the case.

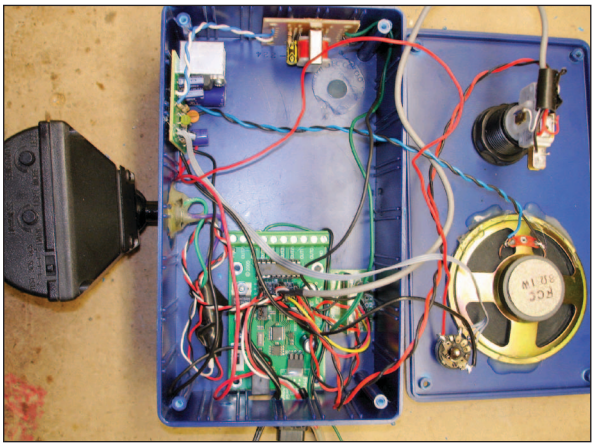
The eight watt amp kit (see Figure 11) was easily mounted in the same manner as the sound module (again, a dab of hot-melt glue) and the output from the eight watt amp kit soldered to the speaker on the inside of the box. I mounted a NO pushbutton switch with an integral LED indicator light (I had this in my junk box, but you could use one of the NO pushbuttons included with the COB sound module kit) to the inside cover of the box, and then soldered another salvaged CD-AUDIO cable to it as a pigtail and connected that to pin 7 of the Prop-1.

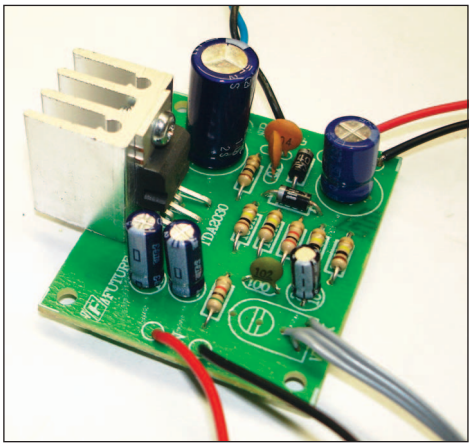
I drilled holes in the box cover to allow the sound to escape from the 3" speaker and then used hot-melt glue to mount the speaker to the inside cover of the box. Lastly, I threaded the power and IR beam break sensor cables through the hole in the side of the box. Now that I had the hardware all assembled, all I needed to do was write some code!

Go With the Flow

During the prototyping, I had written small routines to test each function of the train controller but I hadn't put all the pieces together. Now I needed a final, master program that would encompass all the functions. The software would need to control the train, the sound, read the button, sense the train position, and monitor the dining room for motion.

As the function list got longer, the coding concepts were getting quite complex. If you are ever faced with this situation, I highly recommend that you spend some time building a flow chart for your program as it will really help you to clarify your vision and help to write code to





■ FIGURE 11. Fully assembled audio amplifier.

handle every function.

Figure 12 shows a basic concept flowchart that I built before attempting to actually write code for this project. The program breaks down into these functions:

• Check if the button has been pressed. If so, add laps.

• Check value of the laps variable and stop/start the train.

• Check if the idle time has expired. If so, add a lap.

• Check if the IR beam is broken. If so, decrement a lap.

• Check if the motion is sensed, if so, reset sleep time.

• Increment all time counters.

The program takes quite a bit of space in the Prop-1's memory, but I managed to get all the functions in that I was seeking and still had enough room to add both a diagnostic mode to help align the IR beam sensor and a motion sensor walk test. At the time of this writing, the program provides the following operations:

1) Provide a test/calibrate mode for the IR LED beam.

2) Provide a walk test mode for the motion sensor.

3) Read the button and add a number of laps when pressed.

4) Decrement laps whenever the train passes through the IR beam.

5) If the train is idle for a preset time, then the train runs a single lap.

6) If room is empty for preset time, put the train to sleep.

7) Play a sound when the train starts up (Train Whistle).

8) Play a sound when the train stops (Crossing Bell).

9) Play a sound when the train is passing the IR sensor (Click/Clack).

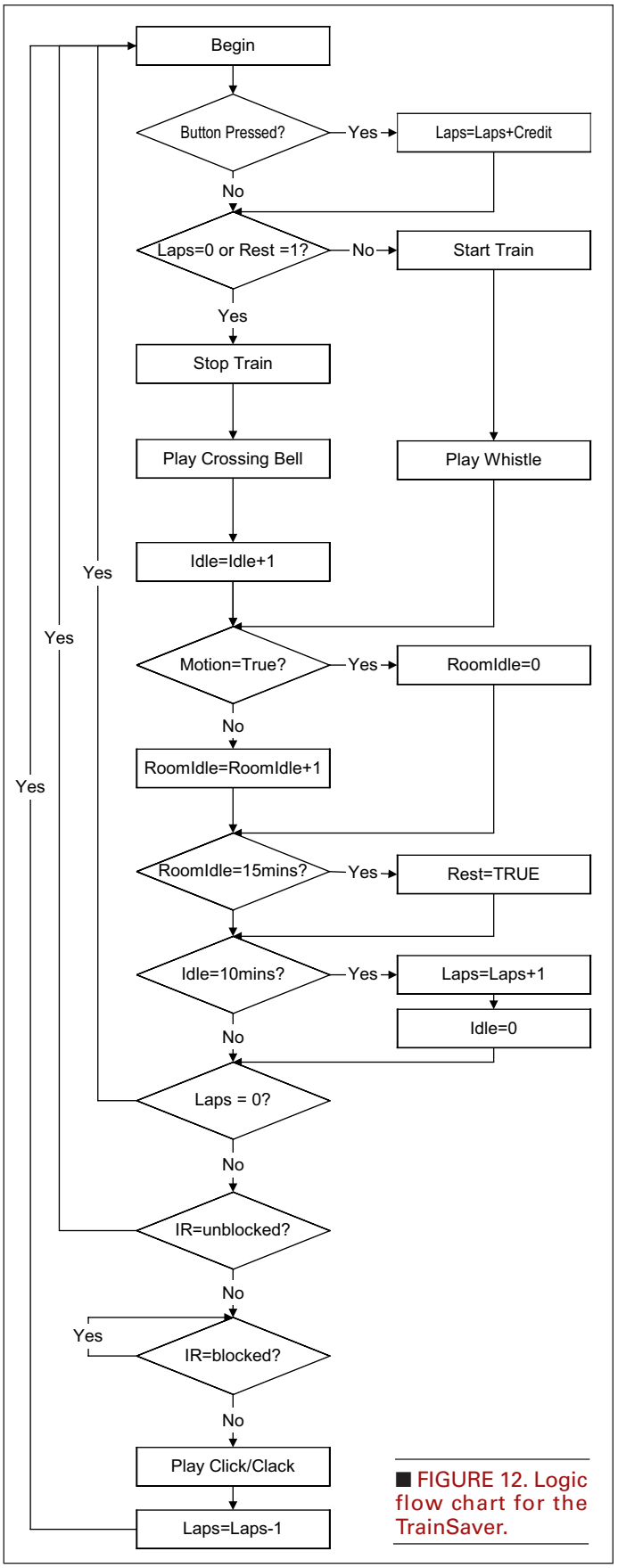
10) Blink the LED to indicate the number of laps remaining.

Testing ... Testing ... is This Thing On?

Now that we had implemented all the functions in a complete program, it was time to set up the system and test it out on my workbench. The first thing needed for testing is power. I used a nine-volt 500 mA filtered power supply to run the microcontroller to insure it had clean power. Make sure the switch on the Prop-1 is set to the 0 position, then connect the transformer to the Prop-1 power connector. Now we need to connect the unit to the train like this:

• Connect the GND wire from the train transformer to the GND connector of the Prop-1.

• Connect the positive wire from



■ FIGURE 12. Logic flow chart for the TrainSaver.

the train transformer to the positive connector for the train track.

• Connect the negative connection on the train track to "output 5" on the Prop-1. (Be careful not to dislodge the jumpers between outputs 5, 6, and 7!)

WARNING: Check the polarity of the train transformer output and make SURE you do not reverse the voltage provided to the Prop-1 screw terminals! Reversing the polarity (by placing the train transformer into reverse) can destroy the ULN-2803 or other components! If you believe there is a danger of this occurring, you might want to place a 1N4001 diode in series with the train power source to block reverse voltage from the circuit.

Turn on the train transformer and set the speed to approximately 50%. Now, turn the switch on the Prop-1 to the 1 position.

WARNING: If you use the 2 position it will attempt to join your wall-wart supply to that of the train power supply! This could destroy the Prop-1 controller!

Once you have the Prop-1 showing power, it's time to download the program. Open the Parallax BASIC Stamp Editor and load the source code (available for download from the website shown at the end of this article). Once the source code is loaded, locate this section near the top of the program shown in Listing 1.

These settings should be examined and understood before changes are made. To better understand these settings, here are some short summaries of the settings and what they do:

TrainIdleTarget: This value is used to determine how long you want the train to wait between automatic laps. The default value is 10 minutes. The idea is to reduce the total amount of runtime on the train engine while still having the train perform often enough to be interesting to observers. If the train were to run continuously, it would not only wear the train out sooner, but people would rapidly lose interest. In a restaurant situation, having the train run on 10 minute intervals means an average patron will see three to six laps during an average meal.

RoomIdleTarget: This value is used to determine how long the performance area of the train remains empty before the train is put to sleep. This further reduces the total runtime accumulated by the engine and also gives the staff a break from the sounds the train makes.

MaskTime: This value is set in milli-seconds and reflects how long you want to ignore the IR beam-break sensor after it is tripped. This value setting is critical if you are unable to set the IR beam sensor so it stays interrupted the entire time the train and cars are present. You will need to set this value to cause the sensor to wait until the entire train has passed the sensor before decrementing a lap.

Credit: This value determines how many laps are added to the train for each press of the button. If the button is replaced with a coin mechanism, this can be used to determine how many laps are given in return for a coin. Most coin mechanisms have a SPDT switch that is used to detect the coin. Simply replace (or wire in parallel) the coin switch with the existing NO pushbutton.

LapLimit: This allows you to limit how many total laps may be added to the train. This value was added to limit how many laps could be added if someone pressed the lap-button repeatedly. In our testing, we discovered that sometimes the staff would rapid-press the button multiple times adding *lots* of laps (i.e., 10 or more rapid presses at five laps each!). This insures that we don't over run the train.

Once you've adjusted these settings, download the code to the Prop-1. If the download is successful, hold down the lap button and press the reset button on the Prop-1. The program checks to see if the lap button is in a down position on startup. If so, it will enter a calibrate mode where it simply illuminates the indicator LED when the IR beam is intact, and then extinguishes the LED when the beam is broken. This allows those of us who cannot see into the IR spectrum to align the LED IR source with the phototransistor sensor across the track.

Now that you can see the

LISTING 1

```
'-----------------------------------------------------
' Initialize Train System Variables
'-----------------------------------------------------
   HIGH LED                        ' Turn off the LED in the button
   SYMBOL TrainIdleTarget = 10     ' Minutes of idle before a lap is added
                                   '  for the train to run (values 0-254)
   SYMBOL RoomIdleTarget  = 15     ' Minutes without motion before the train
                                   '  is put to sleep (values 0-254)
   SYMBOL MaskTime        = 1000   ' How many MS to wait b4 we check for the
                                   '  end of the train after beam-break.
   SYMBOL Credit          = 5      ' Number of laps added by a button press
                                   '  (or coin drop)
   SYMBOL LapLimit        = 9      ' Set the maximum number of laps that may
                                   '   be added
   SYMBOL TicksPerMinute  = 4000   ' 4000 ~number of ticks that pass in a minute
                                   '  Note that if you alter the loop length you
                                   '  will have to recalibrate.
   SYMBOL TunnelDelay     = 6000   ' Delay from end of train till stop execution
                                   ' (allow train to traverse tunnel)
```

alignment of the IR sensor, pick an appropriate position on the track where the train will reliably interrupt the beam when it passes. The mouth of a tunnel or the start of a bridge span are good locations. If possible, I recommend placing the sensor at a height that will keep the beam continuously interrupted between the train cars. You can accomplish this by placing the sensor at the level of the coupler between cars. If this is not possible, you may need to adjust the mask value in the program to allow for all the train cars to pass the sensor before you decrement a lap, otherwise each train car that passes the sensor would decrement a lap!

Once you have the beam aligned, it's time to test out the motion sensor. Press the lap button one time to drop into the walk-test mode. The LED will illuminate each time that motion is detected. Try holding very still and see if the LED goes out. If it does, you know that the motion sensor is working. Now press the LAP button one more time to exit the calibrate mode. Once out of calibrate mode, the system is waiting for either the lap button to be pressed or for enough idle time to pass to cause a lap to be added to the lap counter.

To have the train start, simply press the lap button one time. This will add five laps to the train (the number of laps added by a single button press is adjustable in the software). A single blink of the indicator LED will acknowledge the five-lap credit has been added. The train whistle sound should play and the train should slowly start up and begin to move around the track.

When the train gets to the point where it interrupts the IR beam, the indicator LED on the TrainSaver box should blink the number of laps remaining before the train will stop and the click/clack sound should play. When the laps value has been reduced to 1, and the train breaks the IR beam, the train will slow to a stop and the crossing bell sound should play.

If no one presses the laps button for 10 minutes (or whatever number you set before you downloaded the code), the TrainSaver will add one lap to the lap counter, the train will play the whistle, run one lap, and then halt to the accompaniment of the crossing bell sound. If everything operates as described, then you have a working TrainSaver! Congratulations!

Plug it In, Plug it In ...

Once we completed the bench testing described above, it was time to head out to our BBQ restaurant early on a Saturday morning armed with ladders, tool boxes, drills and all the other bits and pieces we thought we would need. We mounted the TrainSaver box on a wall adjacent to the cash register so the cashier could easily add laps to the box, but placed it high enough on the wall to make it difficult for children to add laps themselves.

We removed the ground lead from the track and ran it into the TrainSaver box and attached it to the Prop-1 screw terminal 5 (being careful not to alter the jumpers between terminals 5, 6, and 7). We then ran a line from the Prop-1 GND screw terminal to the GND connector on the train transformer. I plugged in the wall-wart power supply and turned the power on while holding down the lap button on the unit, thereby placing it in diagnostic mode.

Then, while watching the LED to make sure we had the sensor lined up with the IR LED, we installed the sensor across the tunnel (see Figure 13) making sure the sensor height was at the same level as the engine-to-car coupler. (*Note: This is important to insure the short gap between the engine and the cars doesn't cause extra decrements of laps as each car passes.*) I pressed the button to drop the unit out of diag/calibrate mode and we were ready for our first real test.

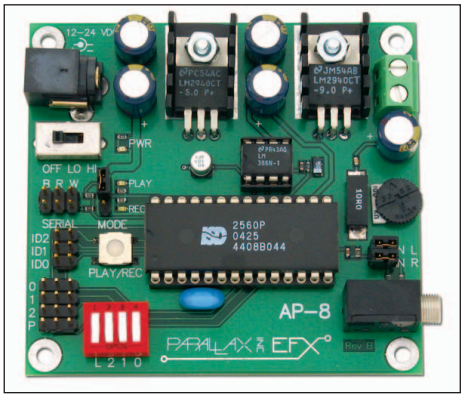
Crock Full O' Trouble!

We pressed the lap button, the whistle sounded, the train engine started up, and the train headed around the track! Success! We then noticed that about halfway around the track, the TrainSaver unit sounded the click/clack sound and the LED blinked to indicate that the unit had decremented a lap. We were surprised since the train was no where near the sensor when this occurred.

When the train did arrive at the sensor, the click/clack played and a lap was decremented as expected. Though puzzled, we chalked it up to an anomalous IR reflection or some other random event that could be ignored. However, as we were packing up our tools, we noticed that the train began to decrement laps every so often when the train was in the middle of the track, sometimes

■ FIGURE 13. The IR beam-break sensors mounted on the train tunnel entrance.

■ FIGURE 14. The ParallaxEFX AP-8 serially controlled digital sound playback unit. If this were used to substitute for the COB sound effects module, it would free up three IO pins on the Prop-1, allowing for expanded functions and more sound effect events.



Author Bio

■ *Vern Graner is a Senior Systems Engineer with a Software Company in Austin, TX. Though married with two kids, he still finds time to create electronic projects in his so-called "spare time." You can contact him via email at vern@graner.com*

causing the train to halt without parking in its assigned location (the tunnel). Something was obviously wrong ... but what?

We double-checked the connections and checked the software but everything seemed to be set up correctly. While I was going over the source code, my son mentioned that a crock pot on a shelf below the train transformer would make a buzzing sound each time that a false lap was detected. With more investigation, we discovered that this crock pot had an old thermostat that was arcing and generating *lots* of noise! It appeared that the interference was being picked up by the IR sensor wire and being interpreted by the software as the train breaking the beam!

Based on this theory, I added a few lines of code that would check the state of the IR sensor and make sure it stayed in a "beam is broken" state for at least 100 samples before it would acknowledge the reading as valid. Once the False Detection code loop was added, the false laps disappeared! Problem solved and the TrainSaver was done!

Future Expansion

Though I'm happy with the current incarnation of the TrainSaver, I've already been brainstorming about future improvements to this device. For example, the pushbutton switch could be replaced with a coin box that would allow patrons to drop a coin to receive a preset number of laps from the train. This would allow the owner of the train system to recoup some of the inevitable cost of repairing or replacing a train engine when it wears out. An external speaker could be placed inside a tunnel or in a fake building near the station where the train stops so the sounds would appear to come from the vicinity of the train.

You could opt to reduce the number of sounds (i.e., remove the click/clack) and dedicate that pin to controlling a relay that would switch the

Resources

■ Vern Graner's "TrainSaver" website
www.trainsaver.com

■ Parallax, Inc.
www.Parallax.com

■ The Robot Group of Austin
www.robotgroup.net

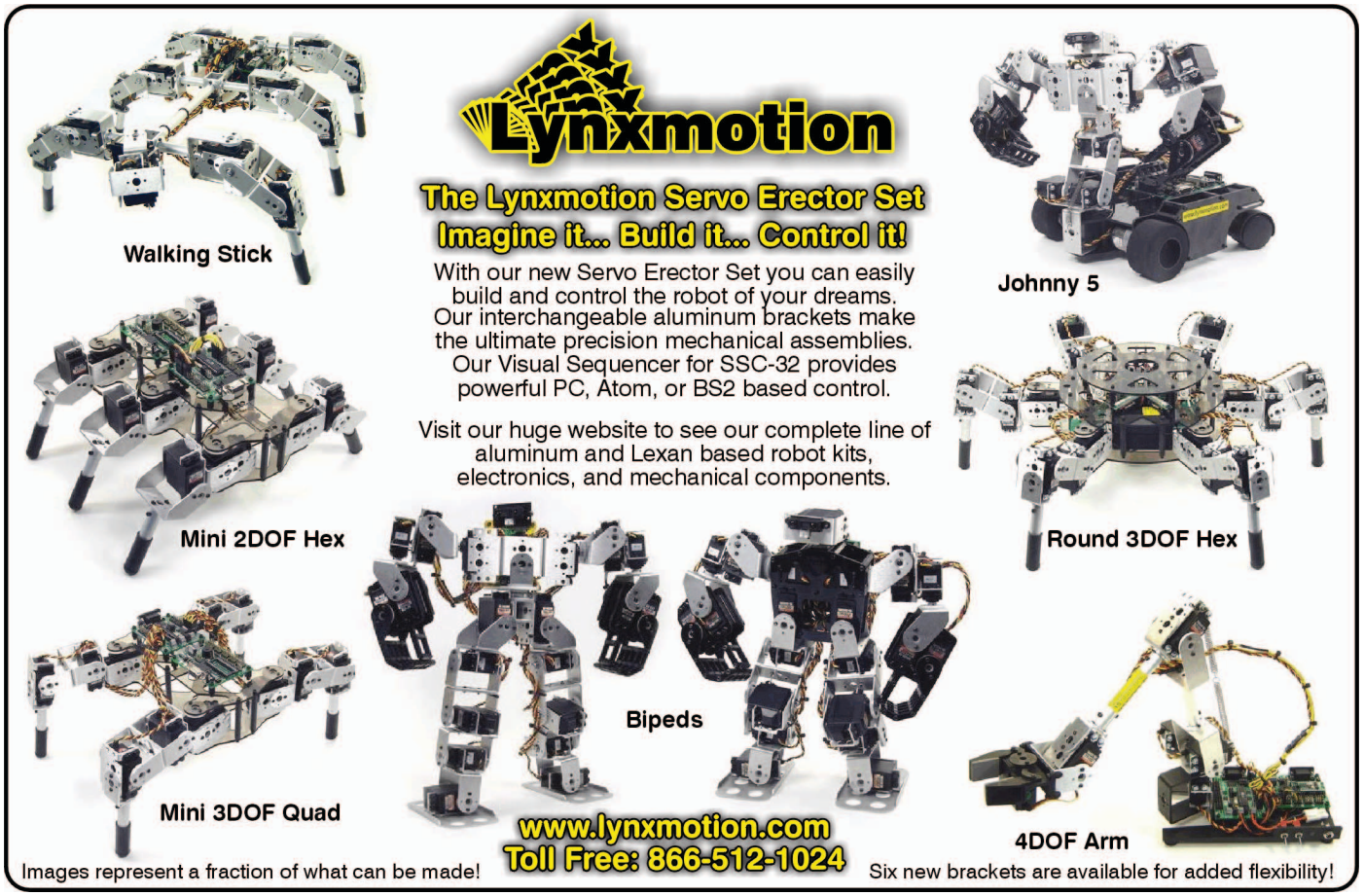
sound between speakers on different parts of the track. You could opt to free up three pins by replacing the COB sound module with the ParallaxEFX AP-8 (see Figure 14) serially controlled audio board. This would let you trigger up to eight sound effects while using only ONE pin on the Prop-1.

Since the configuration of the Prop-1 is already set up for a servo to be directly connected, one could be attached to a pin and code added to control an automatic crossing gate (lift and lower the barrier as the train passes).

It's amazing to me that with such a small and inexpensive controller, there's still so much room for exploration! If you decide to build a TrainSaver, I invite you to visit **www.TrainSaver.com** and exchange ideas, messages, and code improvements with others working on this neat little project. **NV**

PARTS LIST

ITEM	DESCRIPTION	SUPPLIER	PART NO.	PRICE
❑ Microcontroller	Prop-1	Parallax	31101	$35
❑ Serial Programmer	BASIC Stamp 1 Serial Adapter	Parallax	27111	$4.95
❑ Motion Sensor	PIR Sensor	Parallax	555-28027	$7.95
❑ Sound Module	Train Sound Generator Kit	Quality Kits	SG1M	$7.95
❑ Amp	Eight Watt Audio Amp Kit	Quality Kits	FK604	$6.49
❑ IR LED and Sensor	Matched Infrared Emitter and Phototransistor Detector	RadioShack	276-142	$3.29
❑ Project Case	8" x 6" x 3" Project Enclosure	RadioShack	270-1809	$6.99
❑ Case Mount LED	Red LED with Holder	RadioShack	276-068	$2.39

MISCELLANEOUS PARTS

- ❑ NO Pushbutton (Included with the Train Sound Generator)
- ❑ 3" Eight-ohm Speaker (Included with the Train Sound Generator)
- ❑ 50 ohm 1/4W resistor (IR Transmitter LED)
- ❑ 10K ohm 1/4W resistor (Pull up Resistor for Phototransistor)
- ❑ 180 ohm 1/4W resistor (Current Limit Resistor for Case LED)