Hacking a Zip-Zap for Remote Control

by Guy Marsden

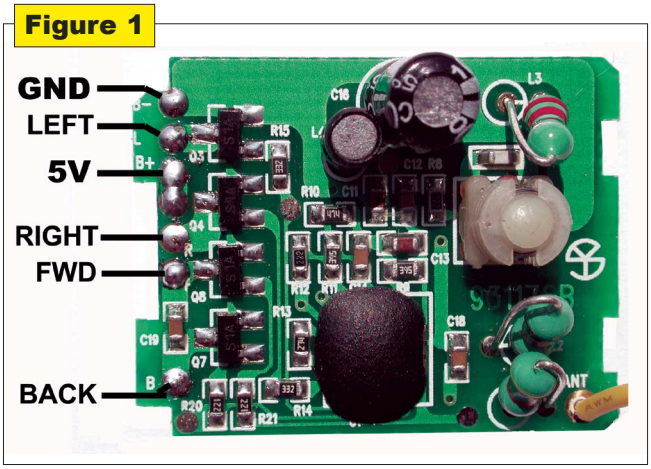
When I was in RadioShack a while back, I noticed that they were selling off the Zip-Zap mini RC toys at deep discounts. The store manager explained that they were closing out the line after a four-year run. (Don't worry — they are still available on eBay and other web resources.) I bought one to indulge my "inner child." I was impressed by the tiny size of the toy car and the fact that it obviously had four channels of control — forward/back and left/right.

After chasing the cat with it for a few minutes and learning that the range was about 10 feet or so, I started to think about applications for a nice, cheap little four-channel control system. A number of ideas came to mind, such as signaling between bots in game playing or flocking behaviors. A transmitter could be used as a sort of lighthouse waypoint for mobile robots that come within its limited range. Or it could be used to keep mobile robots within a certain reception range, such as a specific room.

It could also be used to reduce the number of wires needed to operate manipulators on the end of an arm. Sometimes running multiple wires through a robot arm can be tricky and restrictive; a short-range radio link could reduce a bulky wire harness down to two power leads. So I decided to take my toy apart to see how practical it would be to hack.

Deconstructing the Receiver

Figure 1



On opening up the car, I found a tiny receiver board operating from an equally tiny battery that was marked 1.2V 100 mAH. My first concern was that I would want to get a useful signal level out of the device in order to interface to five-volt logic devices. I removed the battery and clipped in my adjustable bench supply and slowly dialed it up to five volts while operating the transmitter. The car steering (consisting of two tiny electromagnets) still racked back and forth and the motor ran just fine, but much faster. So far so good.

Looking closer at the receiver, I saw a row of SOT-3 parts that were labeled Q3 through Q7, with Q5 and Q8 on the back, so they were likely to be the drive transistors. The solder pads along the edge of the board were nicely labeled: -B, L, +B, R, F, B (see Figure 1). Clearly, these were for power, Left/Right, and Forward/Back. After cutting all the motor wires off close to the board, I soldered some LEDs (with current limiting resistors) to the pads to confirm that the transistors were operating as open collector switches. Success! The transmitter was now controlling four discrete LEDs. (By the way I find it handy to have a bunch of LEDs lying around with 470 ohm resistors attached. They are useful for all kinds of testing and can

be plugged right into a breadboard as an indicator.)

Since the F/B outputs comprise an H-bridge to reverse the motor, it is clear that each output has two transistors apiece that pull the output to the supply rails. This means that a small relay can be connected to one of the outputs and tied to V+ or ground and it will switch when actuated! These outputs can also interface directly to logic since they are driven both ways. The L/R outputs each have a single transistor driver with an open collector so these can switch a relay coil to ground only. To interface to logic, a pull-up resistor would be needed; a 4.7K or 10K will work fine.

Deconstructing the Transmitter

After opening up the transmitter box, I first removed the two AAA batteries (three volts), connected my adjustable supply, and dialed it up slowly starting at three volts while trying the buttons and watching my LEDs blink on the receiver. It worked just fine at 5V, but at 6V it ceased to produce results; the range and response did not seem to be compromised at five volts based on my relative tests. I tried pressing the rubber pads on both the L and R buttons simultaneously hoping to see two LEDs light on the receiver. No such luck — it defaults to just the R output on the receiver. So the system can only operate Left *OR* Right and Forward *OR* Back as I suspected. I was hoping to transmit four bit BCD to make a 16 channel control system, but this way I have the option of transmitting eight discrete values that could be decoded for many uses (See Table 1).

Looking closely at the circuit traces, I noticed that one side of each of the button pads went to ground. I traced the other side of the button pads and found that they all terminated at the 14 pin chip (see Table 2) I attached a color-coded ribbon cable to these points (Figures 2 and 3) and tested them from a CMOS inverter's output to ensure that they would operate the transmitter. This means that interfacing to control logic would simply entail using the output of any logic device or microcontroller pin. It couldn't be a simpler interface.

L	R	F	B	BCD
0	0	0	1	1
0	0	1	0	2
0	1	0	0	4
0	1	0	1	5
0	1	1	0	6
1	0	0	0	8
1	0	0	1	9
1	0	1	0	10

TABLE 1. Transmitter Codes.

IC Pin	CONTROL
1	LEFT
4	FWD
5	BACK
14	RIGHT
3	GND

TABLE 2. Transmitter Connections.

Antennas

It's not a good idea to change the length of the wire antenna attached to the receiver since that would de-tune the RF circuit. However, since the transmitter uses a telescopic antenna, I have to assume that the circuit will be forgiving of adjusting that length. I found by experimentation that anything from a 1" wire to a fully extended telescopic antenna performs about the same. However, as I carried the loose board and battery pack around in my hands, I learned that holding the batteries close to the circuit board (right above or below the components) severely compromised the performance. It would be best to keep batteries and other metal at some distance from a plane parallel to the board; I would think that a metal box would be a bad idea.

Applications

The transmitter board is a tad bulky at 2-1/4" x 3-1/4" since it also incorporates the charging circuit for the car. Since this circuitry is all at one end — near the Forward

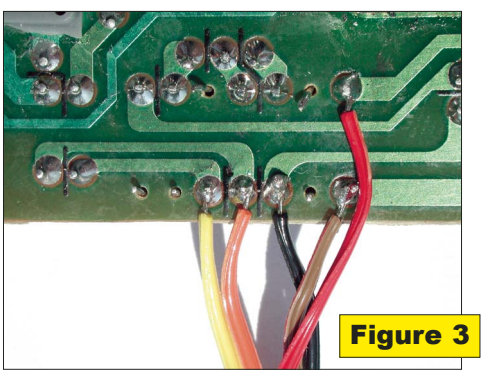
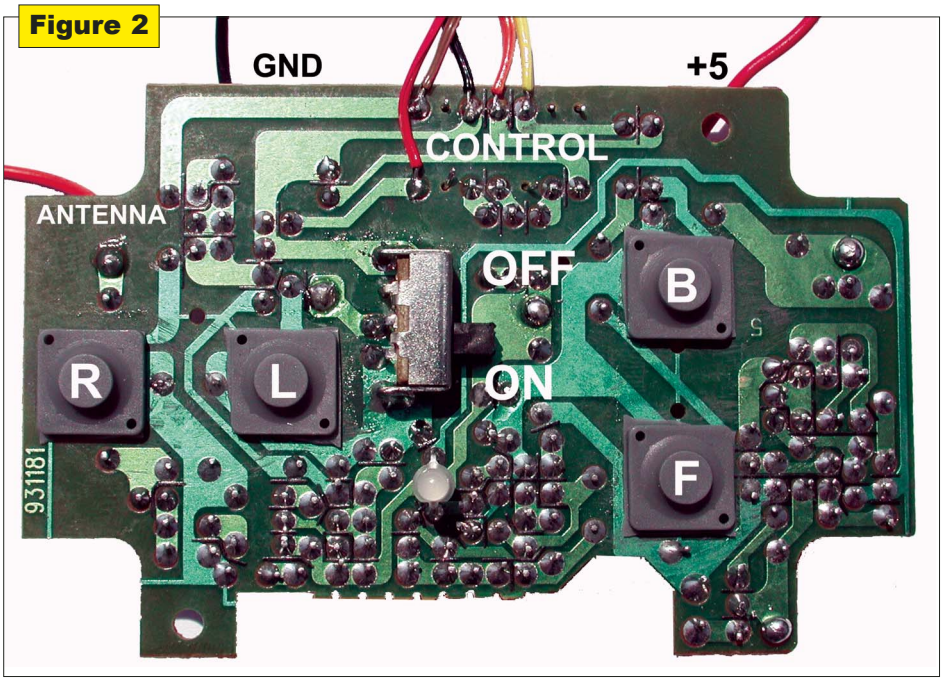
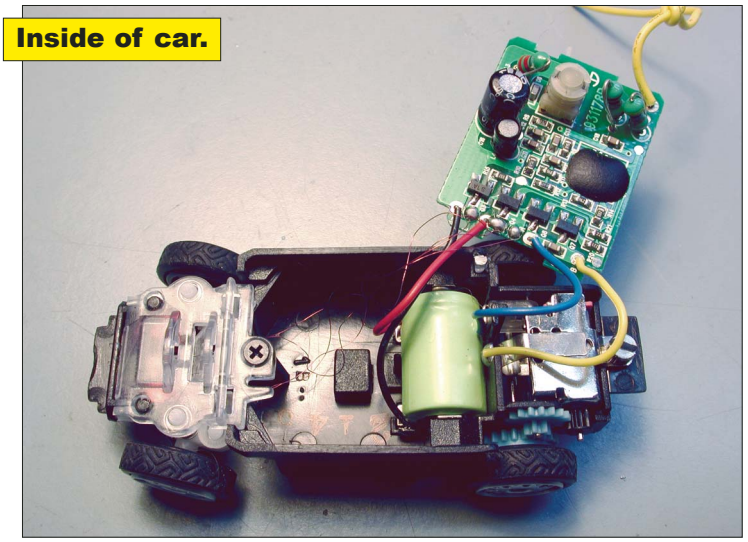
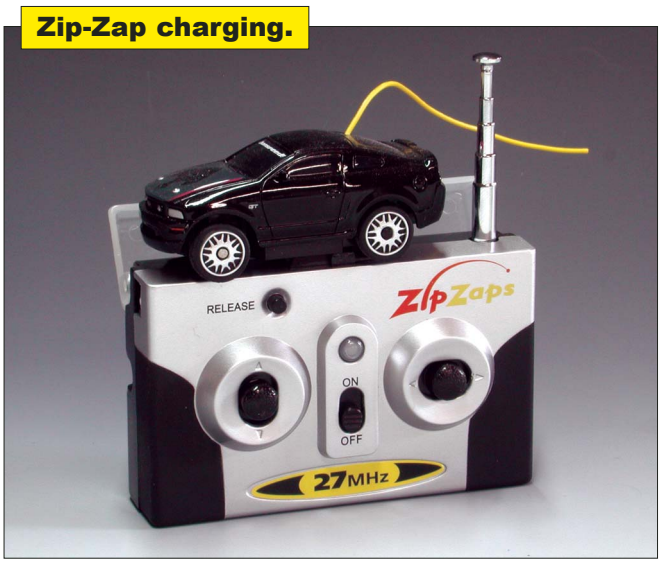


Figure 3

Figure 2







and Back buttons — that part can be carefully cut off if needed. The receiver, however, is pretty darned small at around 1" square and I'm sure that there may be many applications for this tiny four-channel device.

This inexpensive toy can easily be re-purposed in many ways, the applications for the receiver include the following:

- The original use — running one motor forward/stop/back and switching two small loads.
- Switching four small loads or relays.
- Transmitting up to eight discrete codes as a four bit binary coded value.
- A combination of the above like reversing a motor and using the remaining outputs to drive a run/stop relay and a Forward/Back relay for a second motor. **SV**