



約30年前のプラレール「スーパーひかり号」の  
変調信号を解析して遠隔操作する

## HackRF One + GRCでラジコン信号 を解析&エミュレートする実験

野村 秀明

Hideaki Nomura

### 1 概要

微弱無線を使用したラジコンで電車を運転できるおもちゃを解析し、GRC(GNU Radio Companion)上に送信機および受信機を実装しました。また、SDRフロントエンドである“HackRF One”を使用して実際の電波を使用した動作実験を行い、GRC上の送受信機が正しく動作していることを確認しました。

実験に使った環境は下記のとおりです。

- ノート・パソコン：Lenovo ThinkPad X230  
(Core i5 3320M@2.6 GHz, RAM 8 Gバイト)
- OS：Ubuntu Linux 18.04.03 LTS
- GRC Version：3.7.11
- HackRF One Firmware：2017.02.1(API：1.02)

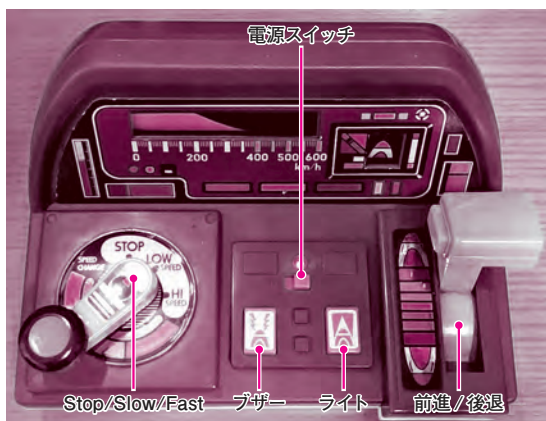
### 2 「スーパーひかり号」の ラジコン信号を解析

私が子供のころ、両親に「ぼくがうんでんするスーパーひかり号セット」という微弱無線で制御する電車のおもちゃを買ってもらいました。マスコン(鉄道車両を運転するためのレバー式スイッチ)を自分で操作

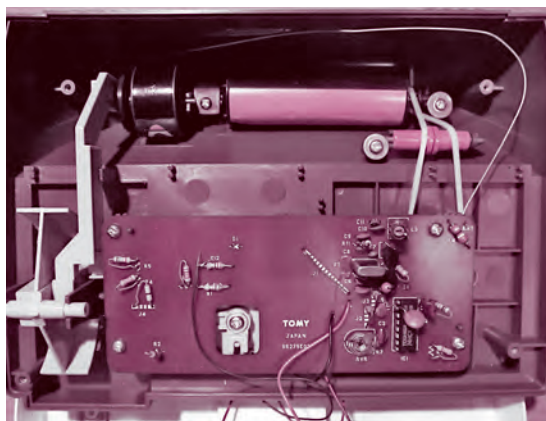
して車両を運転できることは格好よく、夢中になって遊んだ覚えがあります。それからおよそ30年経過して自分にも子供ができ、予定通り(?)電車が大好きな子に育ちました。電車のおもちゃ一式は実家に保管されていたので、自分の子供にも使わせてあげたいと、自宅に引き取ってきました。30年ぶりに電源を入れましたが、スイッチの不良で動作しなかったので、修理するついでに分解して無線部も調査しました。

マスコン型コントローラ(写真1)の中を確認すると、驚いたことに基板上(写真2)にはマイコンらしきものはなく、唯一載っていたICはTC74HC123APという単安定マルチバイブレータだけでした。このICだけでどうやって制御しているのだろうかと技術的興味湧いてきたので、詳しく調べることにしました。

単安定マルチバイブレータは抵抗とコンデンサで構成される積分回路の時定数に応じて出力がON/OFFする時間が決まります。コントローラのボタンが押されたり、スピード・レバーが操作されたりすると、コントローラ内部の接点が摺動します。接点は積分回路の抵抗を切り替えていて、マルチバイブレータの出力信号のON時間が変化します。図1は実機から書き起こした回路図です。低コストでよく考えられた回路だ

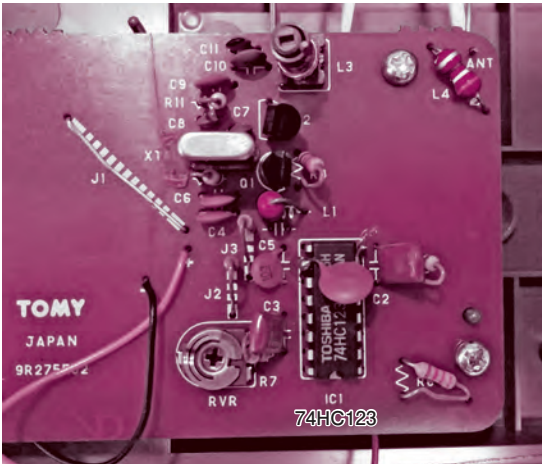


(a) 外観



(b) 内部のようす

(写真1) 「スーパーひかり号」のマスコン型コントローラ

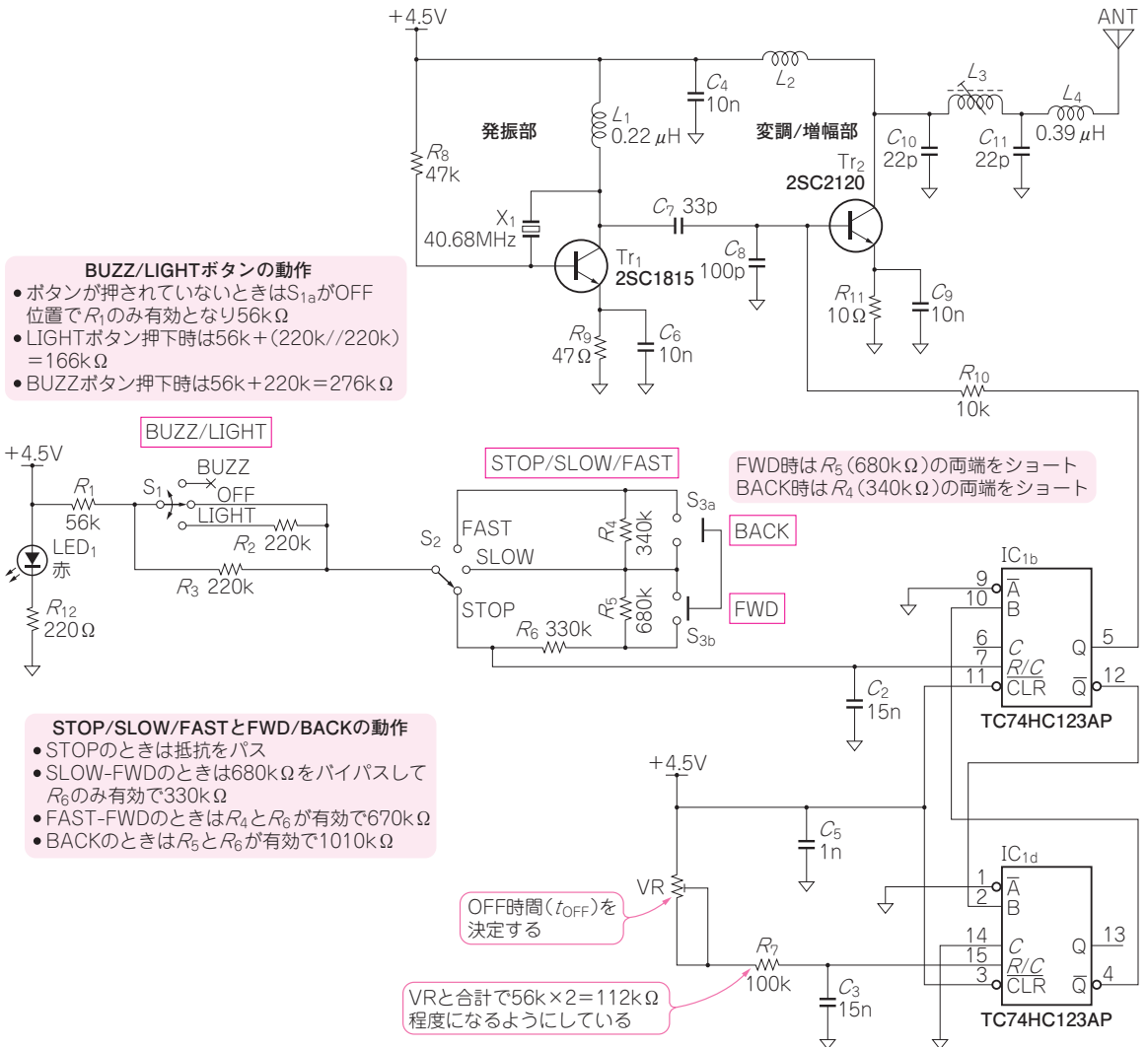


〈写真2〉 マスコン型コントローラの無線制御基板

と思います。

マルチバイブレータで生成されたON/OFFシグナルは $Tr_2$ の変調/増幅回路に接続され、その出力をON/OFFキーイングしてASK変調(OOK)をかけています。水晶発振周波数は40.68 MHzで、ラジコン用の微弱無線局の周波数です。

実際の動作を確認するため、コントローラが出力するRF信号をオシロスコープで確認しました。スピード・レバーがStop位置のとき、図2に示すように $t_{ON} = 0.84\text{ ms}$ 、 $t_{OFF} = 1.68\text{ ms}$ となっていました。 $t_{OFF}$ はおよそ1.7 msで、どの状態でも変わりません。スピード・レバーをSlowにすると図3のように $t_{ON} = 5.8\text{ ms}$ になります。Fastにすると $t_{ON} = 10.9\text{ ms}$ 、進行方向をBackにすると $t_{ON} = 15.9\text{ ms}$ となりました。Backのスピードは1段階だけです。



〈図1〉 マスコン型コントローラの現物から描き起こした回路図