The HamRover: A Vehicle for STEM Education

Terence C. Paddack

KF5TOK

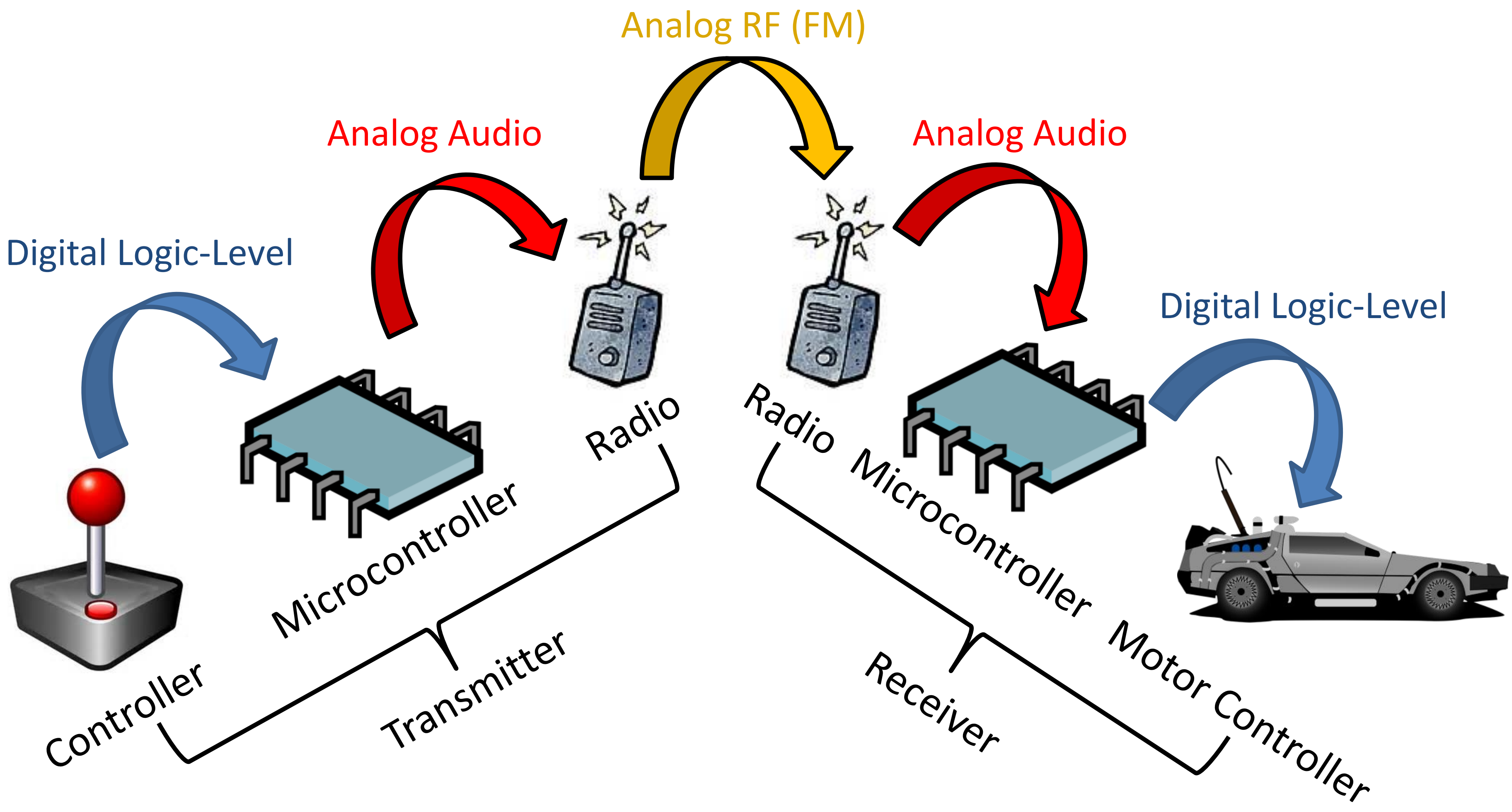
STEMania @ WCWC

Science
Technology
Engineering
Mathematics

Ham radio, anyone?

STEMANIA
SCIENCE • TECHNOLOGY • ENGINEERING • MATH

HamRover Concept - Overview



The HamRover in Action

Legality & Part 97 Compliance

§97.215 Telecommand of model craft.

An amateur station transmitting signals to control a model craft may be operated as follows:

(a) The station identification procedure is not required for transmissions directed only to the model craft, provided that a label indicating the station call sign and the station licensee's name and address is affixed to the station transmitter.

(b) The control signals are not considered codes or ciphers intended to obscure the meaning of the communication.

(c) The transmitter power must not exceed 1 W.

Considerate Operation

Band and Frequency Selection:

- 6m has designated channels for Radio Control
- Inexpensive and readily available radios use VHF or UHF
- 2m and 70 cm regional band plans usually designate frequencies for low power FM simplex or experimental modes
- Texas VHF FM Society Band Plans at http://www.txvhffm.org/coordination/bandplan.php
- Monitor frequencies prior to use to ensure that you will not interfere with QSOs or automated systems

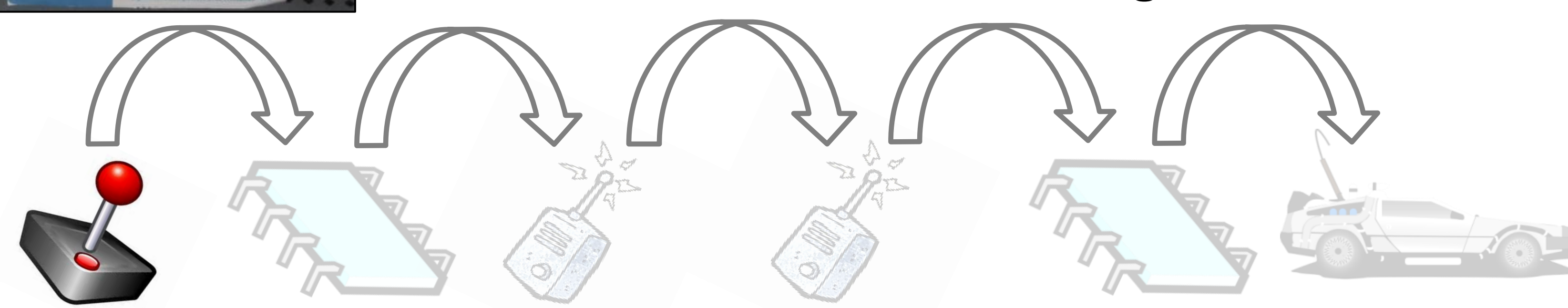
Transmitter - Controller

Nintendo Wii Nunchuck (after-market):

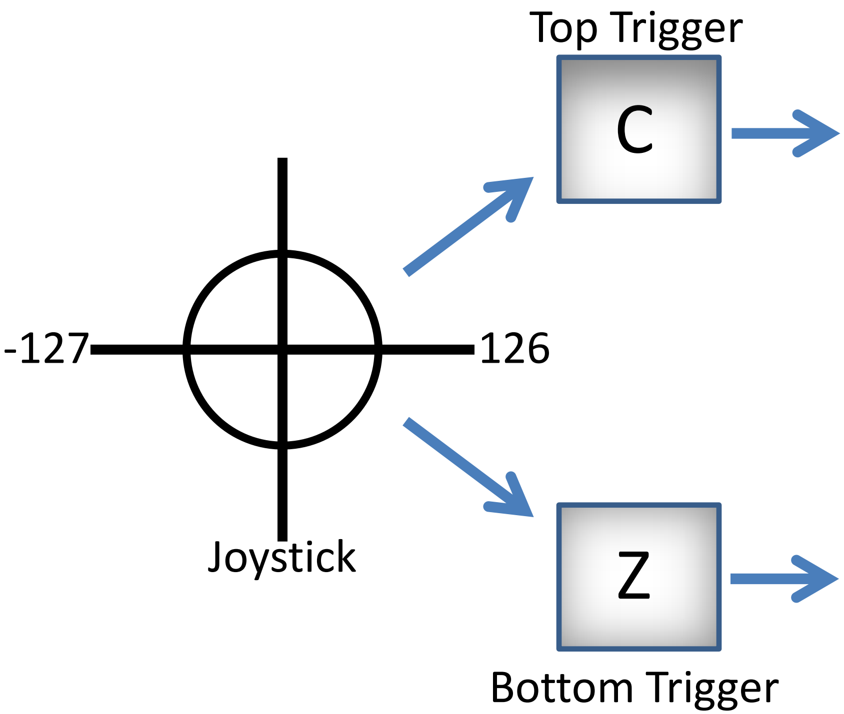
- I²C Data Communication

```
 _________
| 1  2  3 |        1 - SDA
|         |        3 - VCC
| 6  5  4 |        4 - SCL
|_-----_|          6 - GND
```

- C Button (Top Trigger = Forward)
- Z Button (Bottom Trigger = Backward)
- 3-axis Accelerometer (built-in, unused)
- 2-Axis Joystick
  - Y-Axis = unused
  - X-Axis = Left & Right



Transmitter – Nunchuck Data



Direction	Control	% Speed
Forward	300	100
	290	90
	280	80
	270	70
	260	60
	250	50
	240	40
	230	30
	220	20
	210	10
Backward	200	100
	190	90
	180	80
	170	70
	160	60
	150	50
	140	40
	130	30
	120	20
	110	10
	100	0

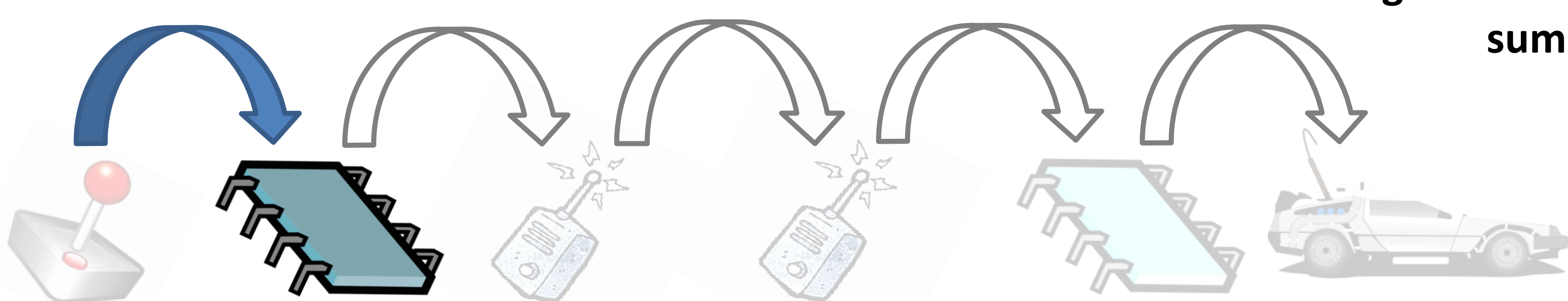
Example Control Code Sequences

Forward (Full Speed)			
!	**300**	**300**	**600**

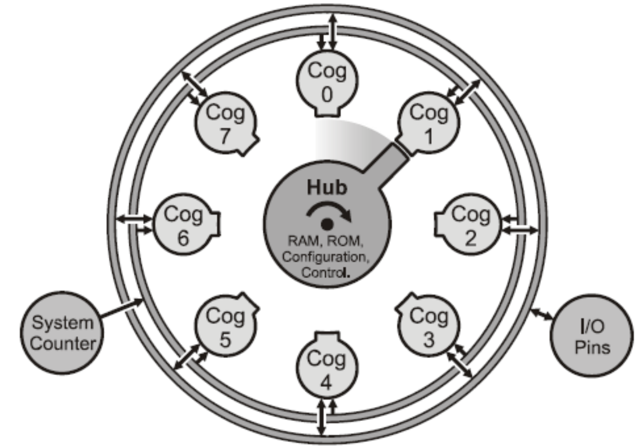
Reverse (Full Speed)			
!	**200**	**200**	**400**

Turn Left			
!	**250**	**300**	**550**

Spin Right (½Speed)			
!	**250**	**150**	**400**
AT	**Left**	**Right**	**Check sum**

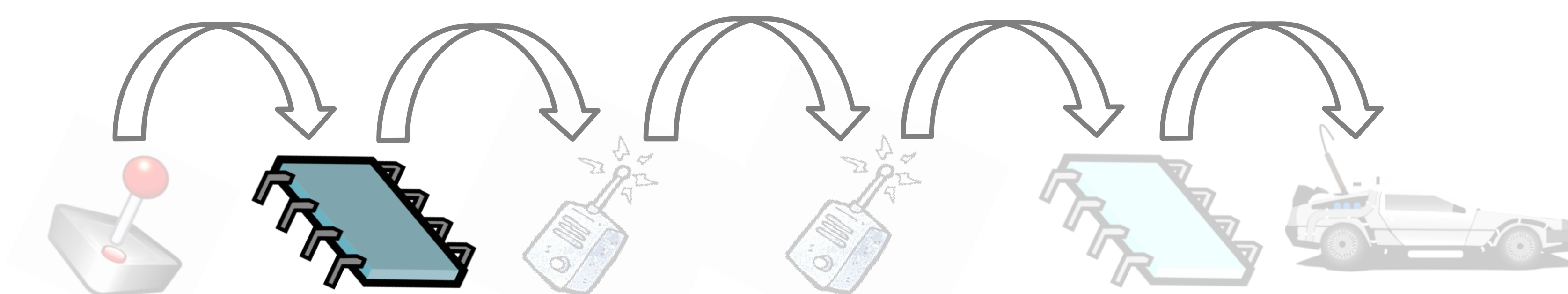


Parallax Propeller Microcontroller

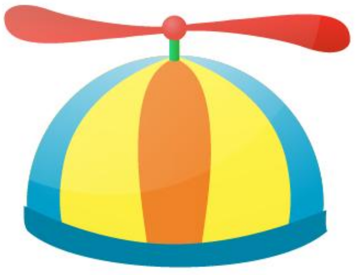


An 8-core (cog) 32-bit IC with:

- 32 IO Pins, 64K ROM, 32K RAM
- Each cog has:
  - 512 (32 bit) longs of RAM
  - 2 highly-configurable, hardware-based, counter modules that can be used for:
    - Analog-Digital and Digital-Analog Conversion
    - Waveform and PWM Generation
    - Edge-Detection & Frequency Counting

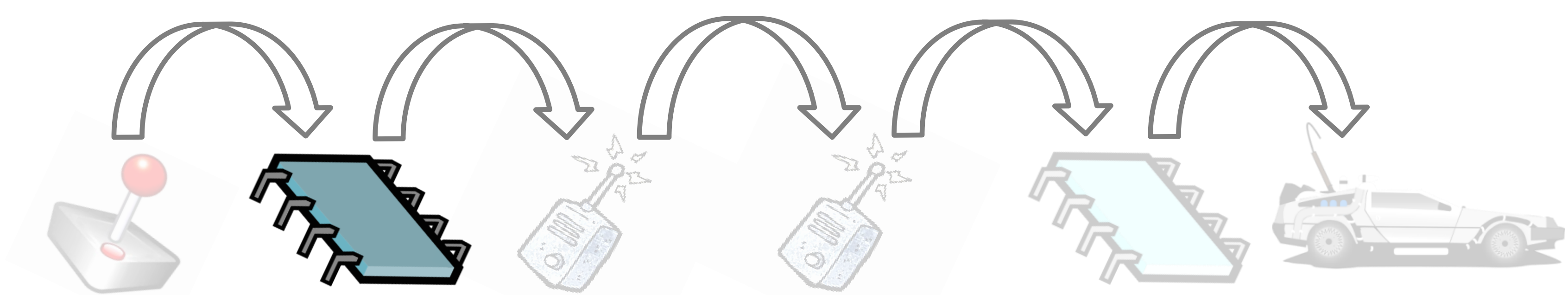


Parallax Propeller Microcontroller



Programmable in multiple languages:

- Propeller Assembly
  - Fastest execution, ‘deterministic’ timing
- Spin
  - Human readable, access to low-level operations
- Propeller C
  - More familiar to experienced programmers
- BlocklyProp
  - Graphical User Interface for beginners (WYSIWYG)

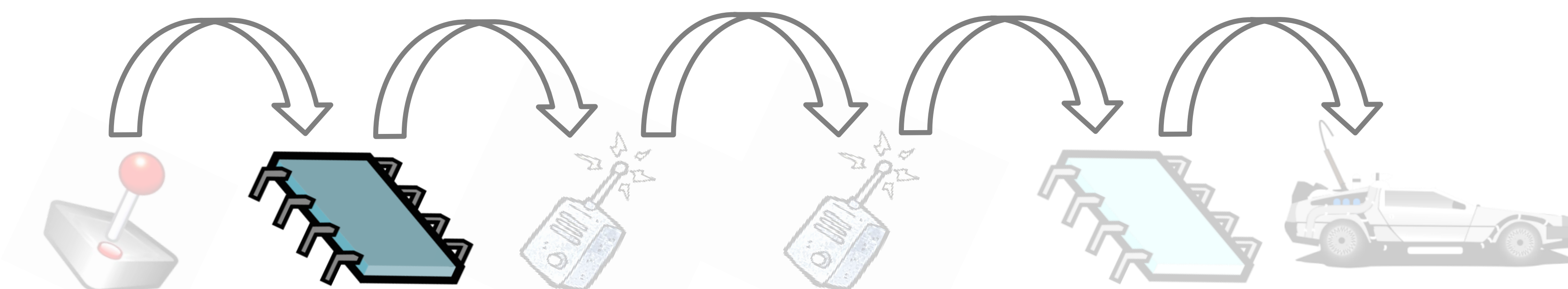


Parallax Propeller Microcontroller

Support

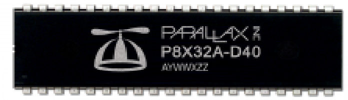
- Hundreds of pages of (free) searchable PDFs in product documentation pages
- Tutorials at http://learn.parallax.com
- Forums at http://forums.parallax.com
- Object Exchange at http://obex.parallax.com
  - Open-source, MIT Licensed code
    - Bell 202 Modem Emulator & Wii Nunchuck Interface
- Open source design (for FPGA programmers)
- Educator discounts available!

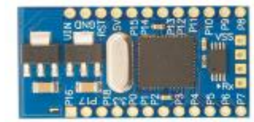
Parallax Propeller 1 open source hardware

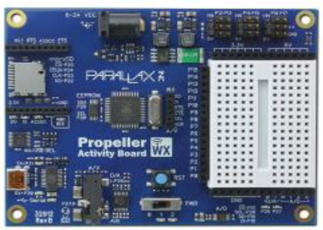


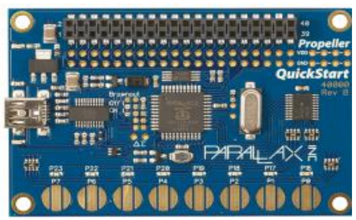
Parallax Propeller Microcontroller

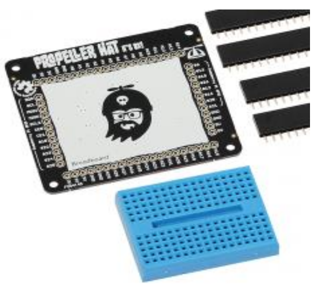
Available in a variety of form factors!

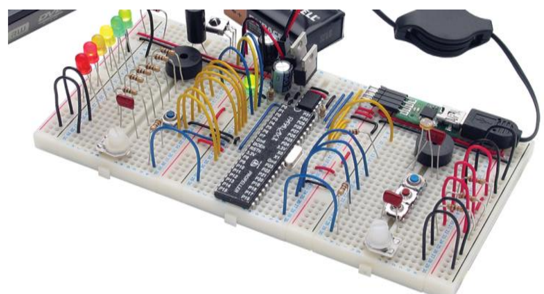


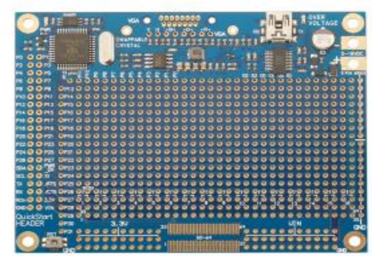


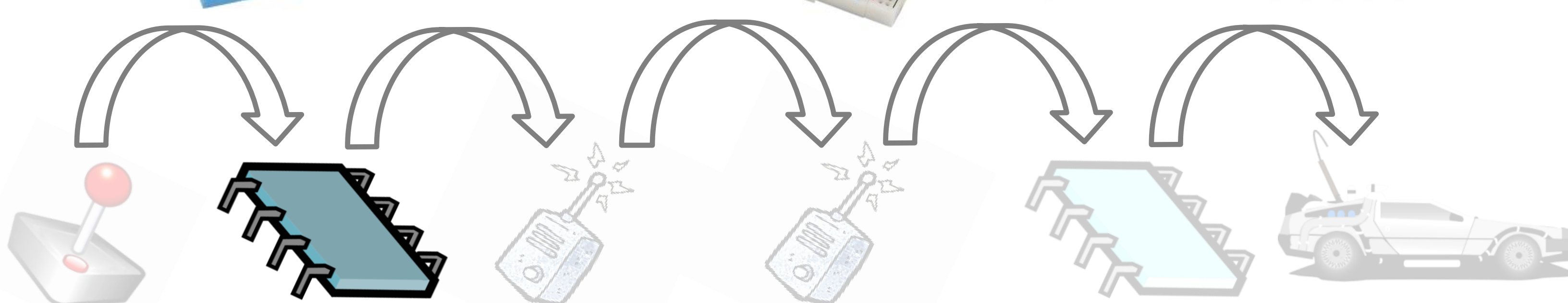




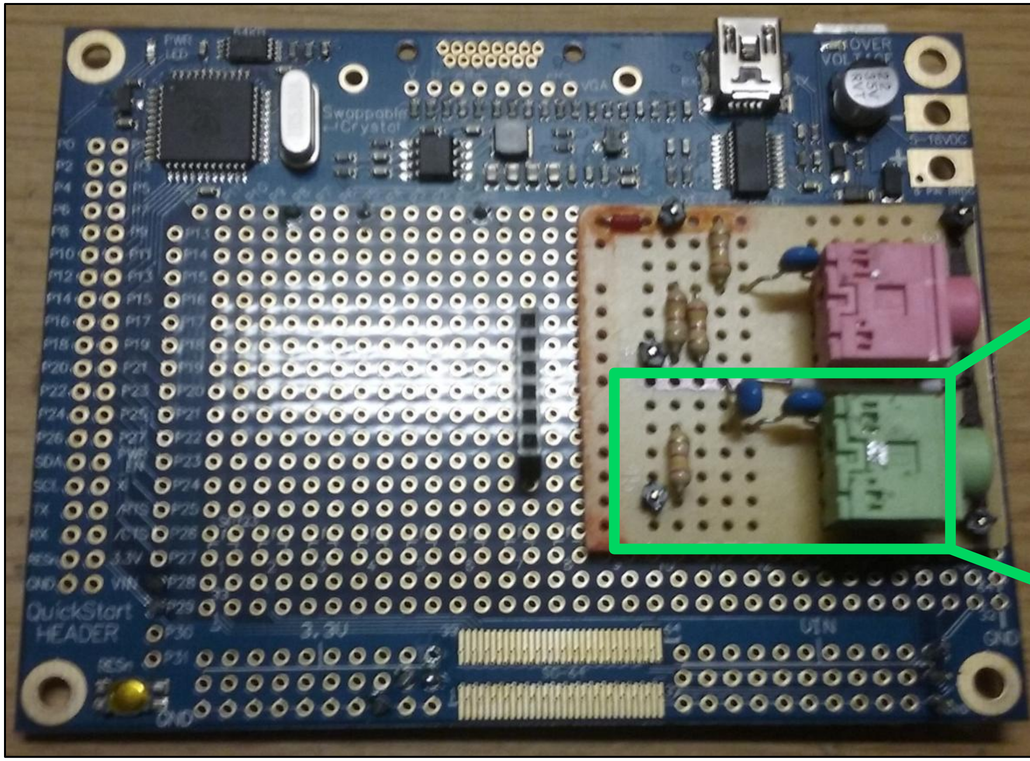




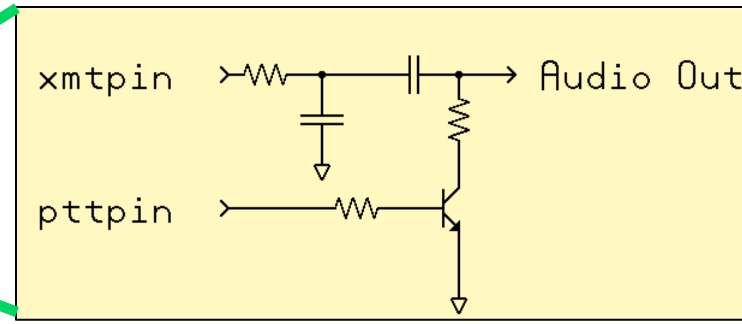




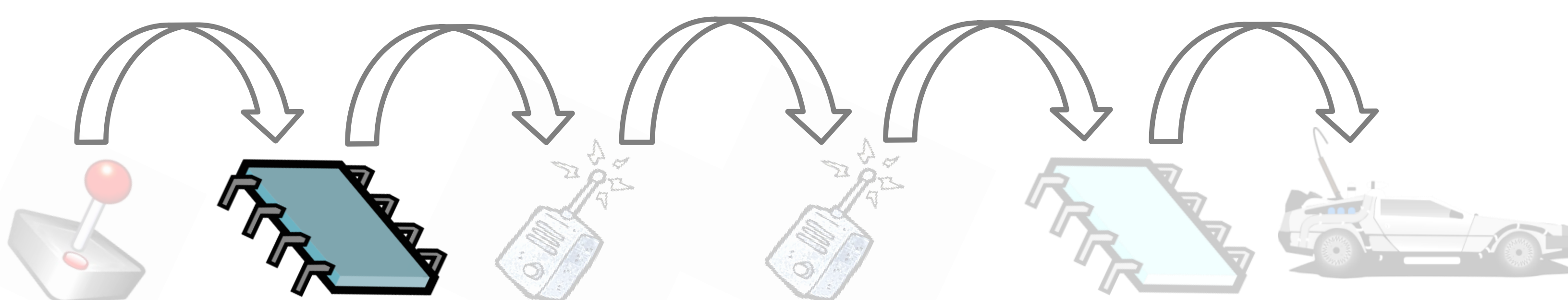
Transmitter - Microcontroller



- Produces 1200 baud AFSK signal by emulating a Bell 202 Modem
- Similar to AX25 “packet” but does not use bit stuffing or NRZI encoding

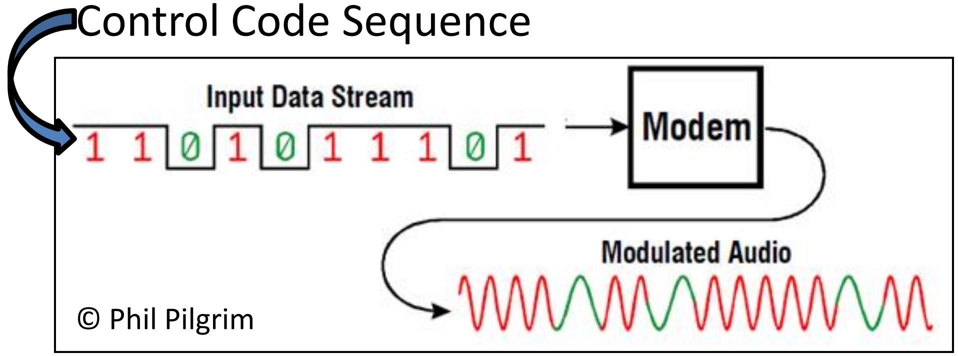


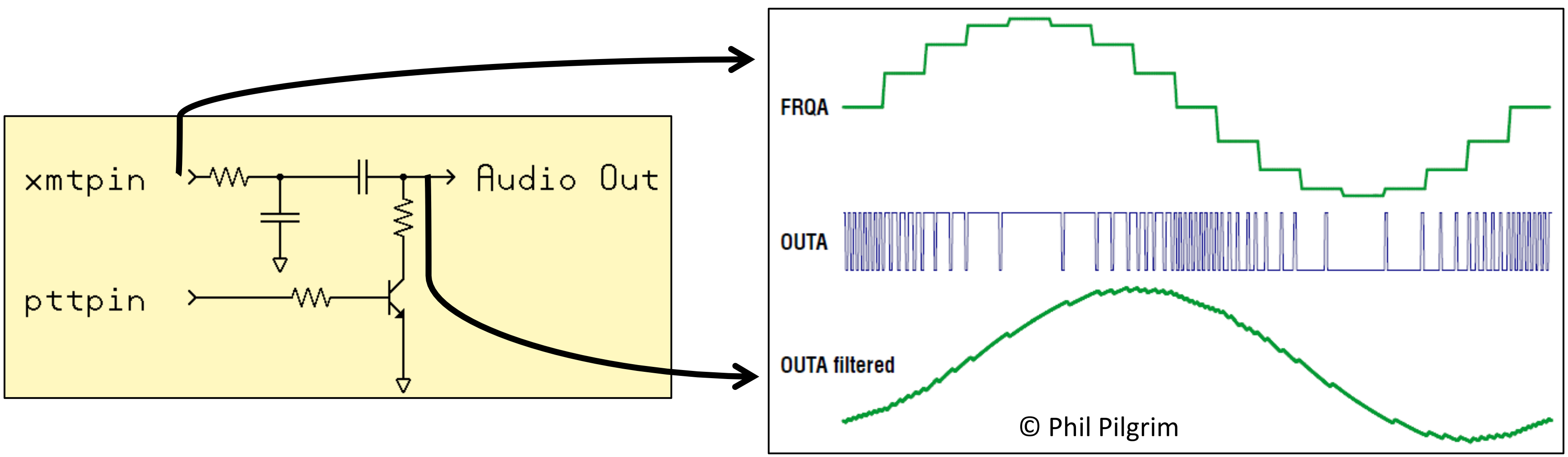
All resistors: 2.2 KΩ
All capacitors: 0.1 µF
Transistor: 2N3904

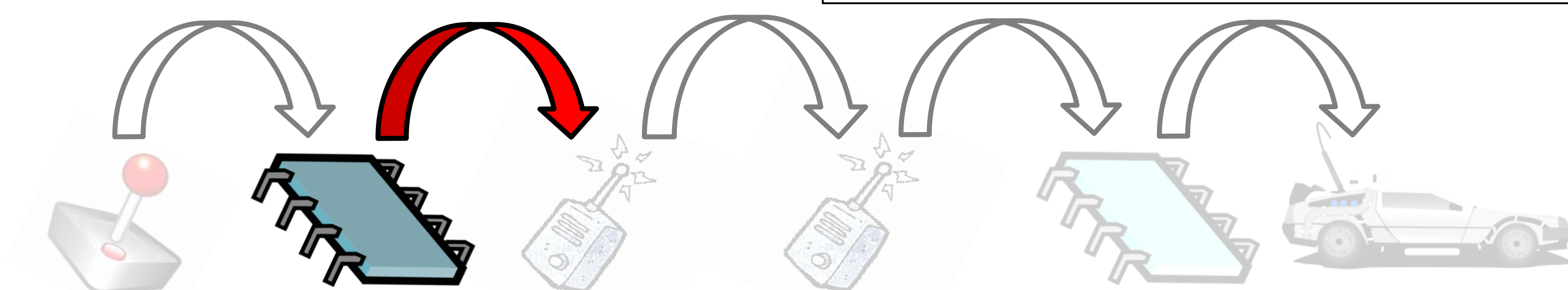


Software - Modulation

Control Code Sequence







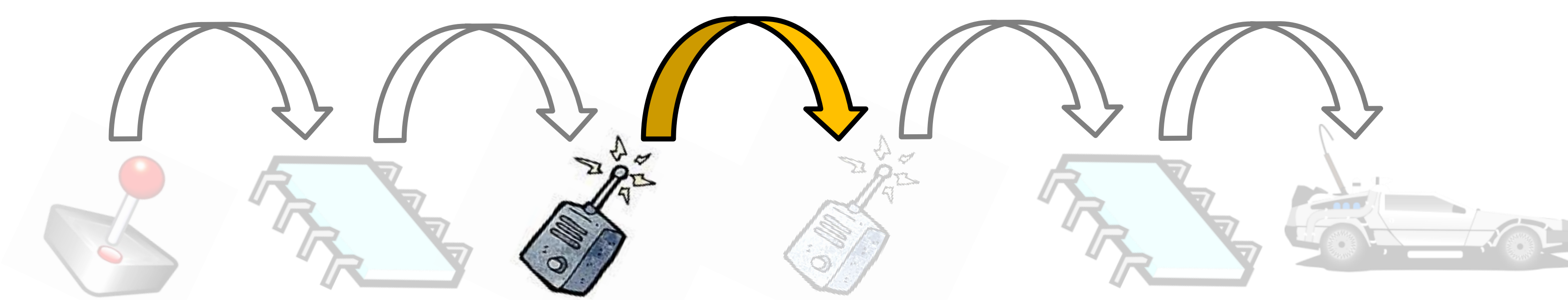
Transmitter - Radio

Baofeng UV-5R$^+$

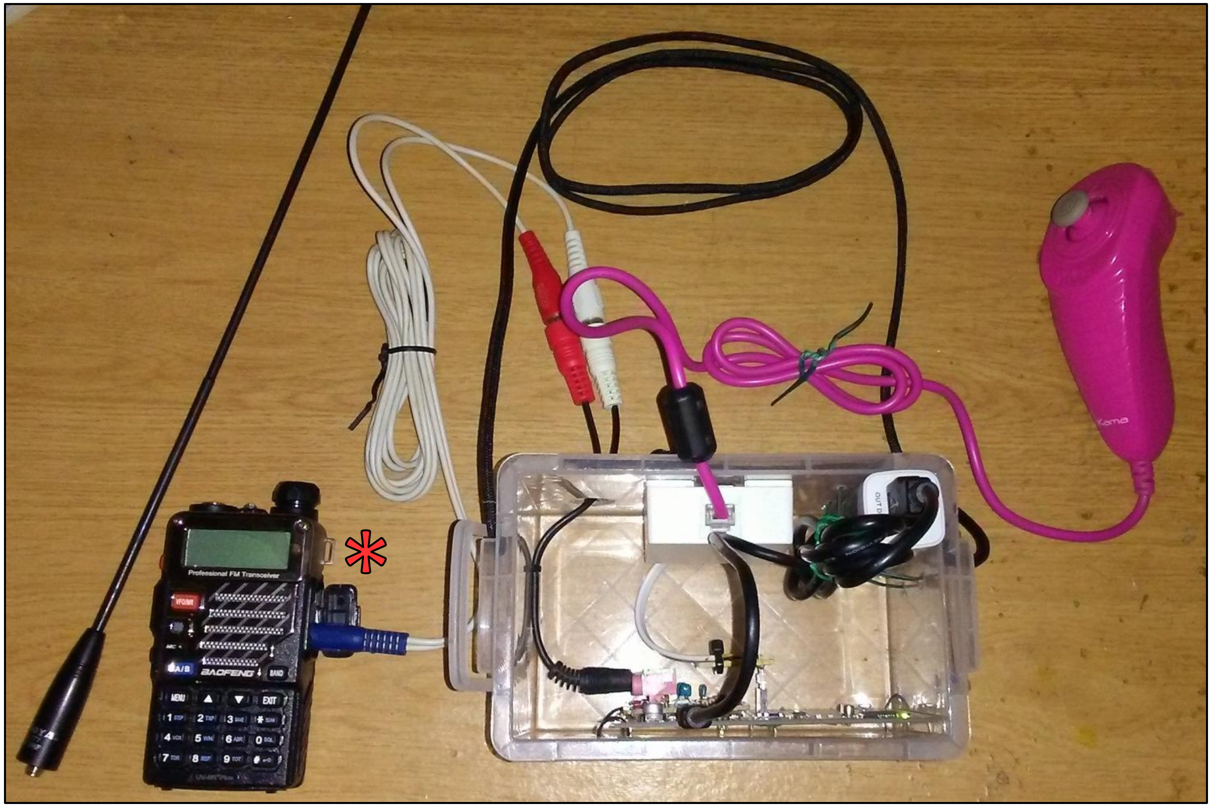
UV-5R$^+$

UV-5RMHP

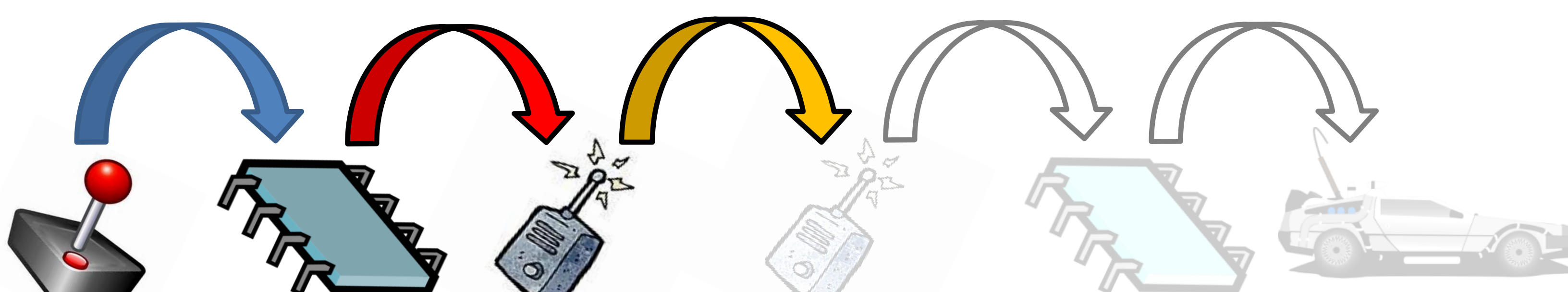
- Settings:
  - VOX: On
  - Low Power setting: 1 W
  - FM Simplex on 2m or 70 cm
  - Transmit Overtime Timer: 600 seconds
  - CTCSS can be used in Tone Squelch is enabled on receiver
- Capable of continuous transmission with brief interruptions every 10 minutes
  - Reliable battery life > 4 hours
- Modem output feeds into 3.5 mm (1/8”) mini mono jack



Assembled Transmitter

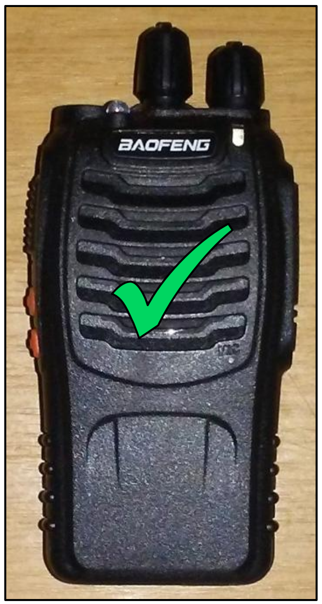


*Don’t forget to label your transmitter for ID purposes!



Receiver - Radio

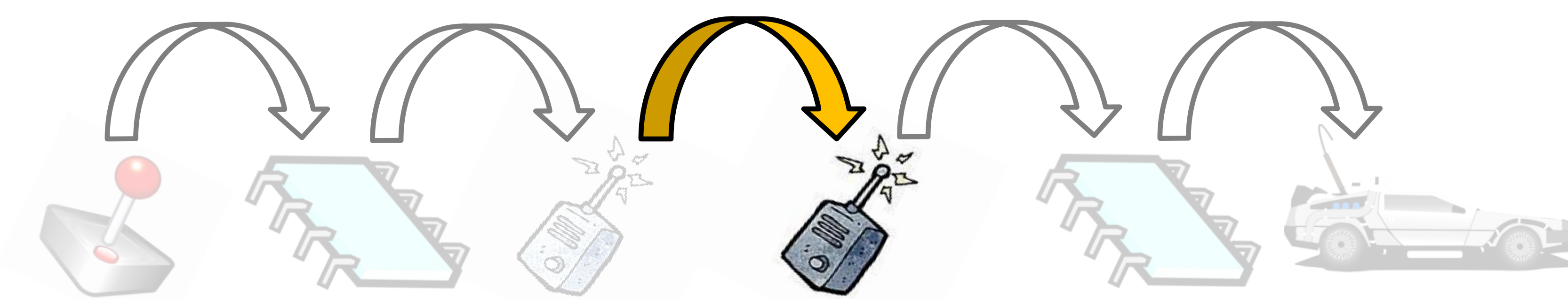
UV-5R$^+$



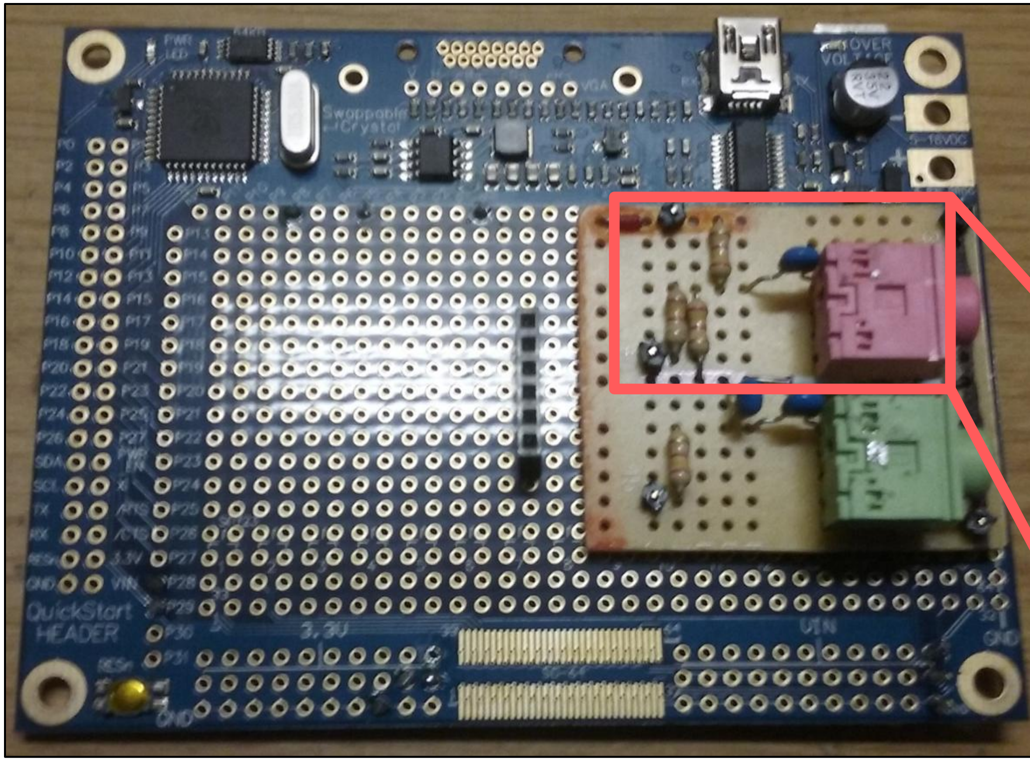
BF-888

Baofeng UV-5R$^+$

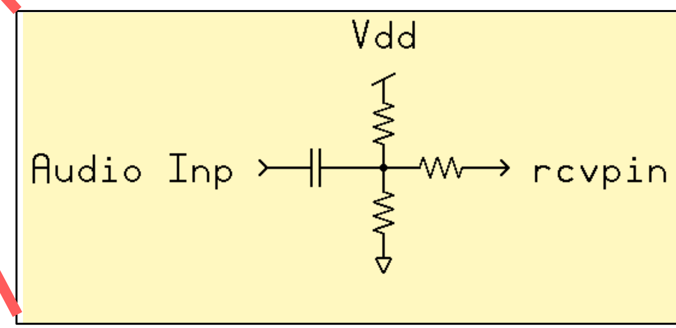
- Settings:
  - VOX: Off
  - Low Power setting: NA
  - FM Simplex on 2m or 70 cm
  - Transmit Overtime Timer: NA
  - Tone Squelch if CTCSS is enabled on transmitter
- Modem input fed by 2.5 mm (~3/32”) sub-mini mono jack
- BF-888 also works, but only on 70 cm



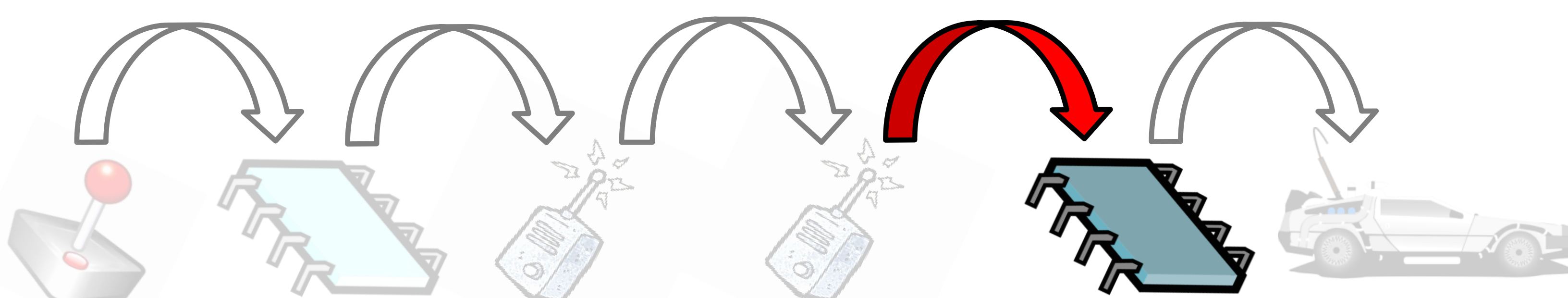
Receiver - Microcontroller



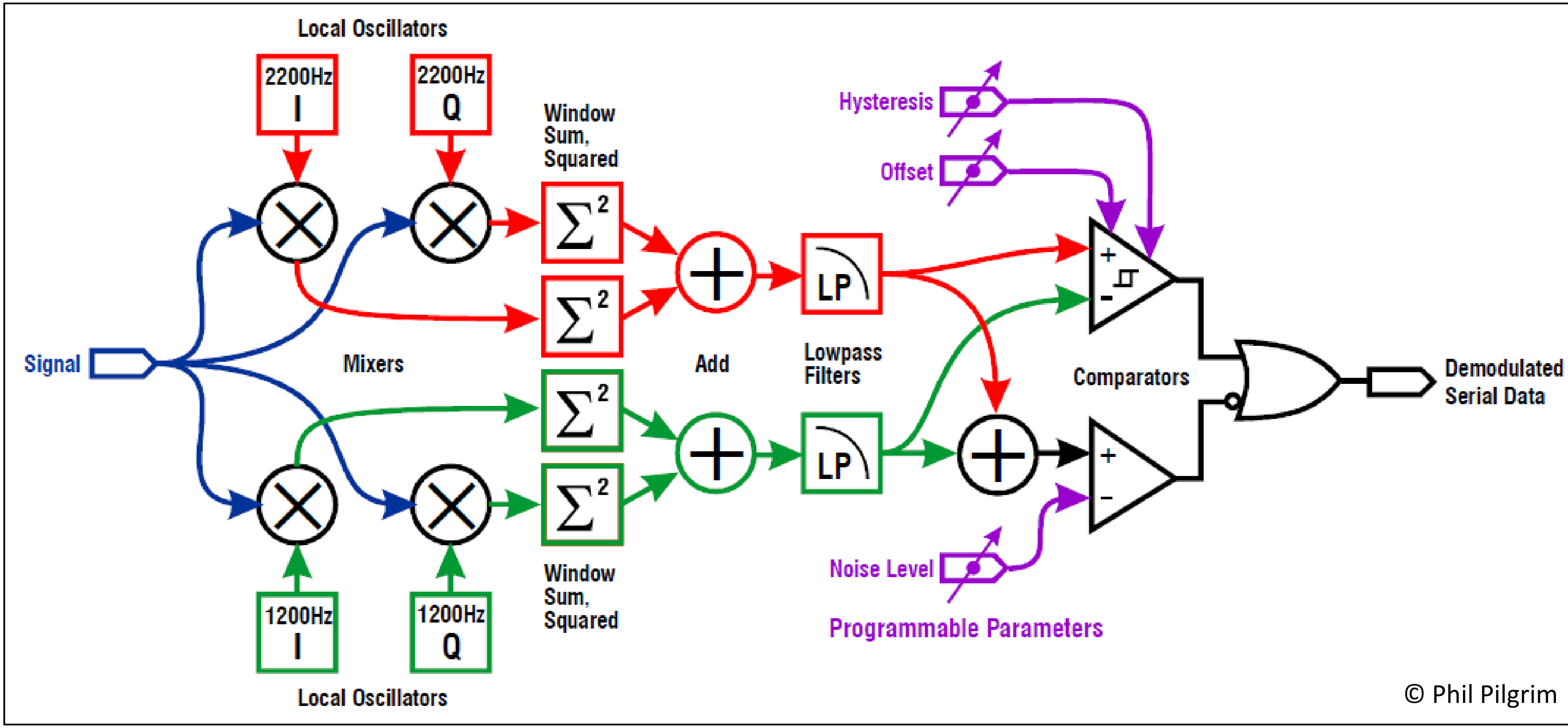
- Demodulates AFSK signal by emulating a Bell 202 Modem
- Returns a string containing the original control code sequence

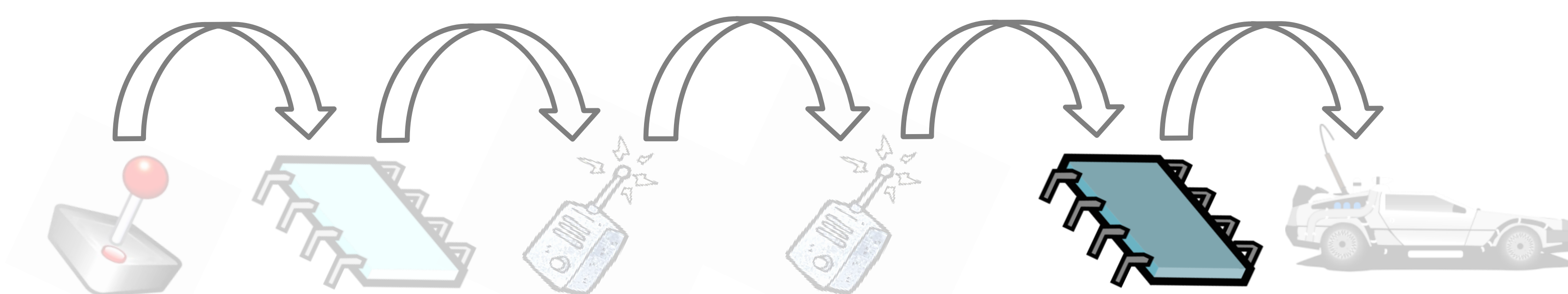


All resistors: 2.2 KΩ
All capacitors: 0.1 µF

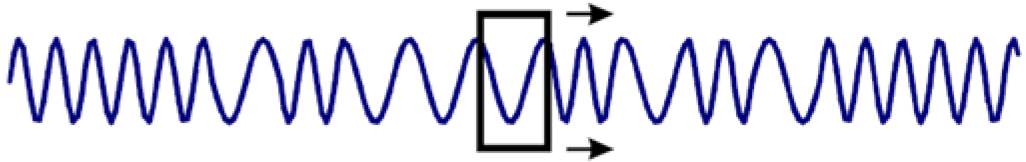


Software - Demodulation





Software – Demodulation

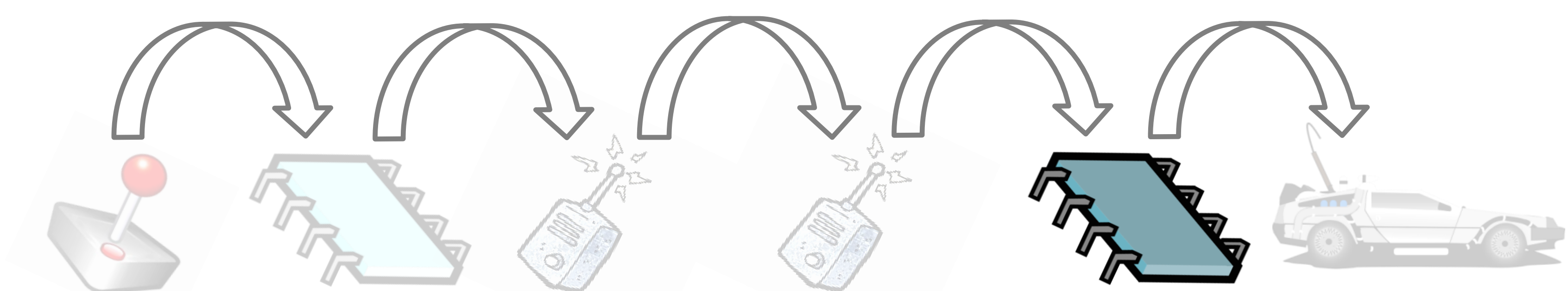


At each position, we will analyze the waveform inside the window for its 1200 Hz and 2200 Hz frequency components and gauge their relative amplitudes. Windows in which 1200 Hz dominates will be deemed "more zero than one", and *vice versa* for windows in which 2200 Hz dominates.

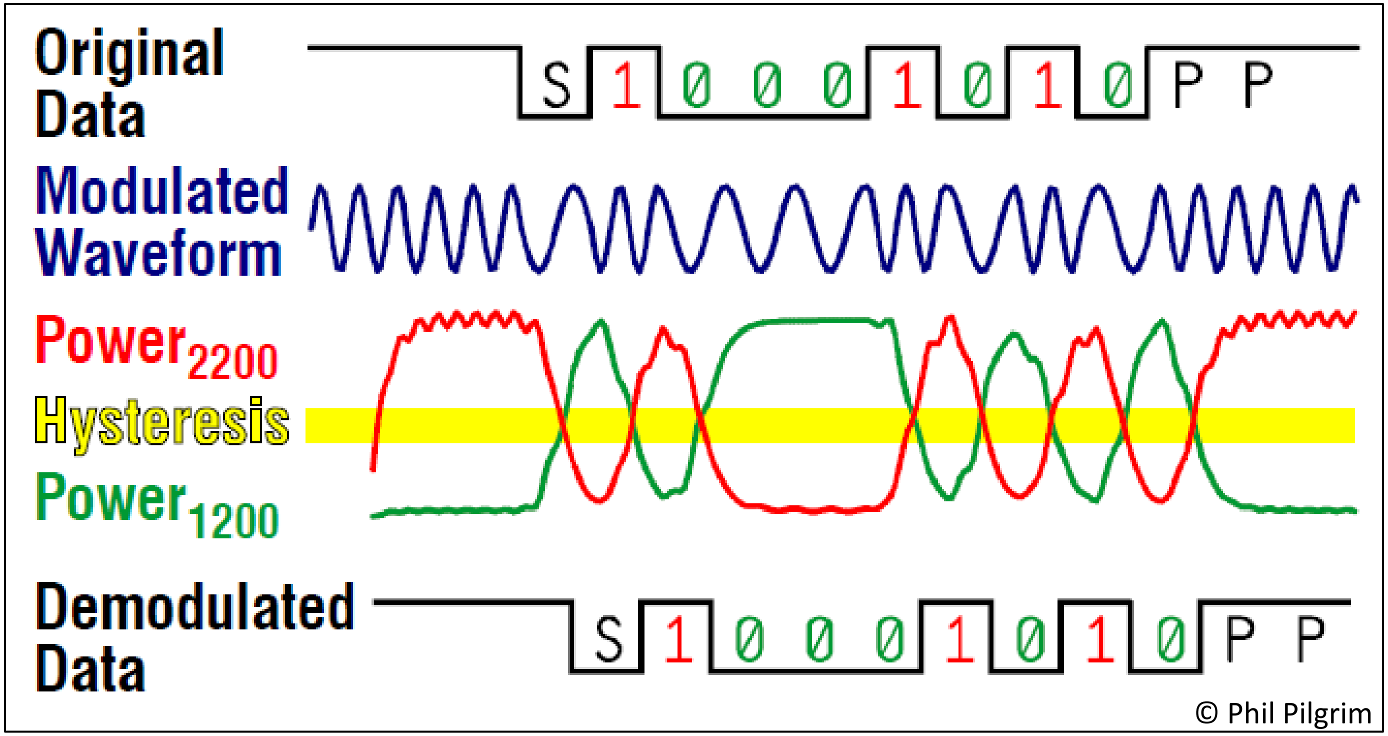
To get these relative amplitudes at each window position, given the incoming modulated signal samples $\mathbf{Sig_i}$, we will compute the *Fourier power coefficient* for each frequency: 1200 Hz and 2200 Hz. This is given by:

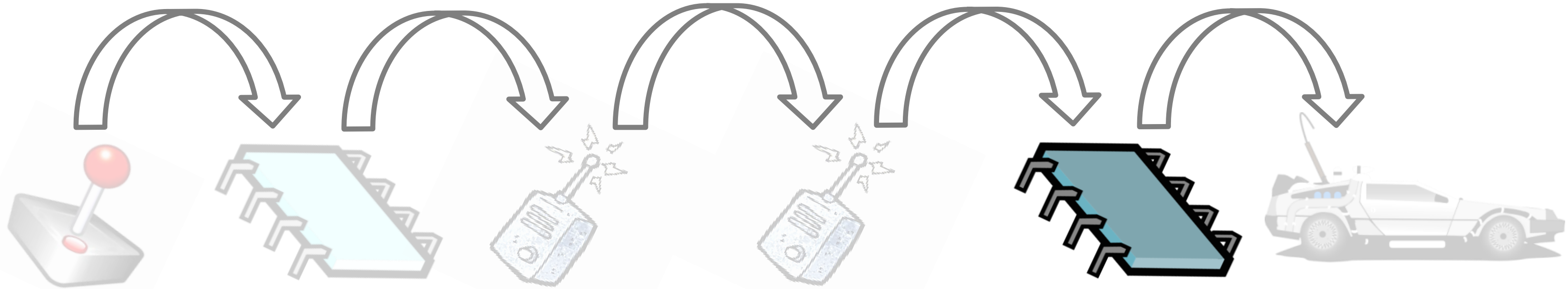
$$\mathbf{Power_f(t)} = \left[\sum_{i=t-15..t} \mathbf{Sig_i} \cdot \sin(2\pi \cdot i/16 \cdot f/1200)\right]^2 + \left[\sum_{i=t-15..t} \mathbf{Sig_i} \cdot \cos(2\pi \cdot i/16 \cdot f/1200)\right]^2$$

By comparing $\mathbf{Power_{1200}(t)}$ with $\mathbf{Power_{2200}(t)}$ at each incremented window position $\mathbf{t}$, we will be able to tell whether the waveform fragment inside the window is "more one than zero" or "more zero than one", and we can assign a corresponding binary value to that window position.



Software - Demodulation

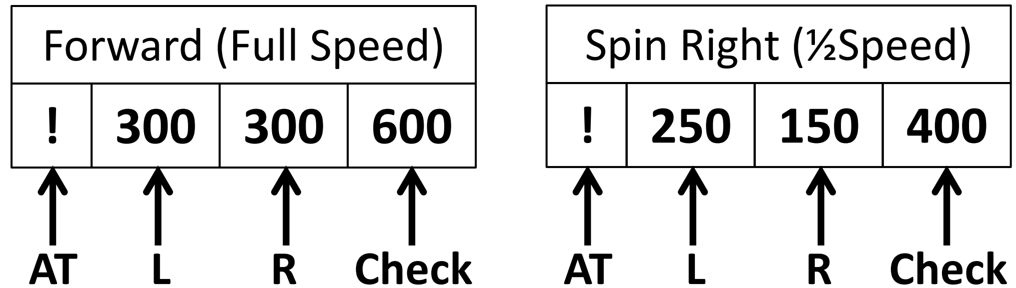


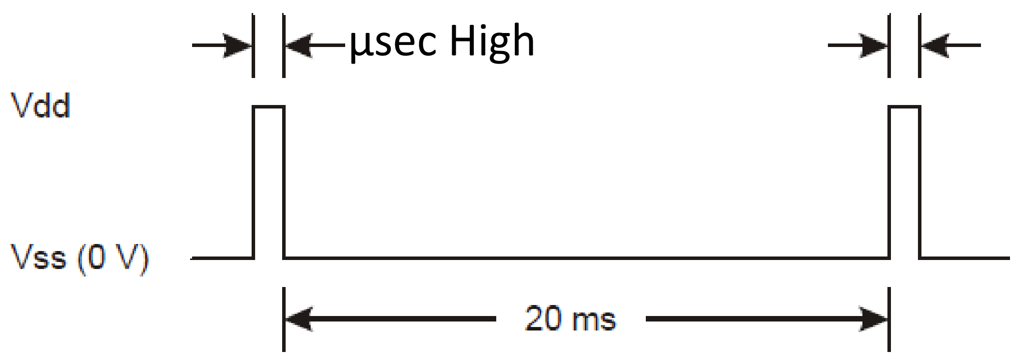


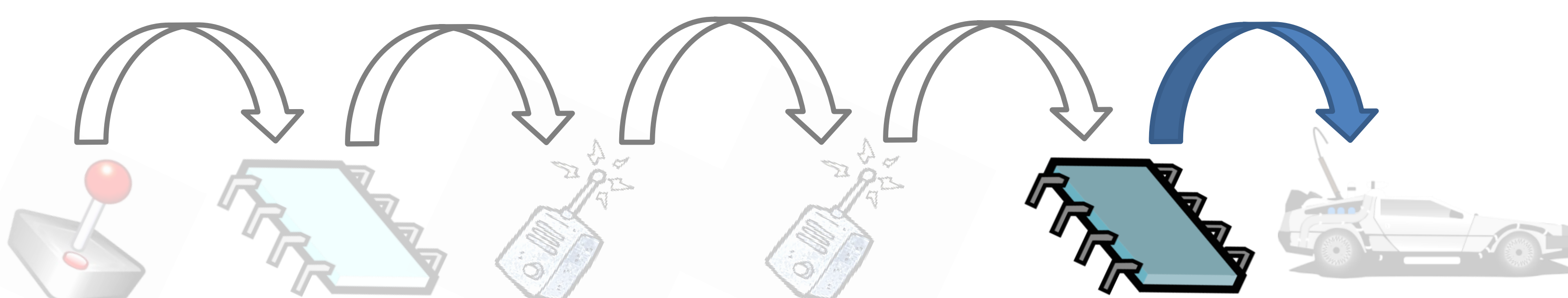
Software – Controlling Motors

Direction	Control	% Speed	µsecHigh
Forward	300	100	2000
	290	90	1950
	280	80	1900
	270	70	1850
	260	60	1800
	250	50	1750
	240	40	1700
	230	30	1650
	220	20	1600
	210	10	1550
Backward	200	100	1000
	190	90	1050
	180	80	1100
	170	70	1150
	160	60	1200
	150	50	1250
	140	40	1300
	130	30	1350
	120	20	1400
	110	10	1450
	100	0	1500

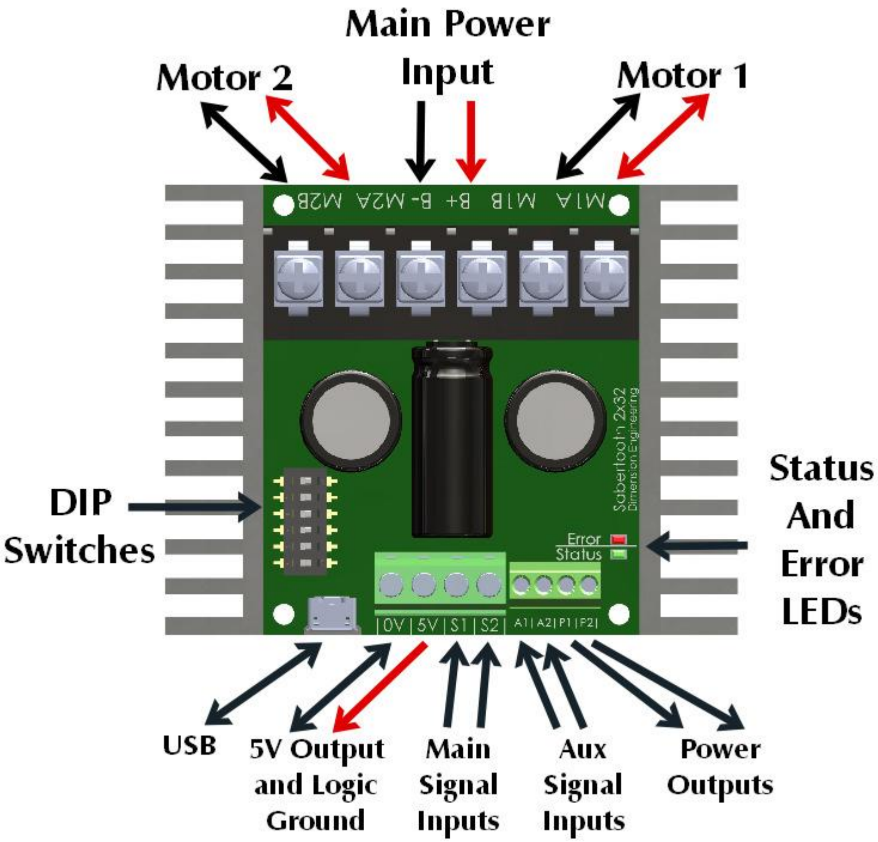
Example Control Code Sequences





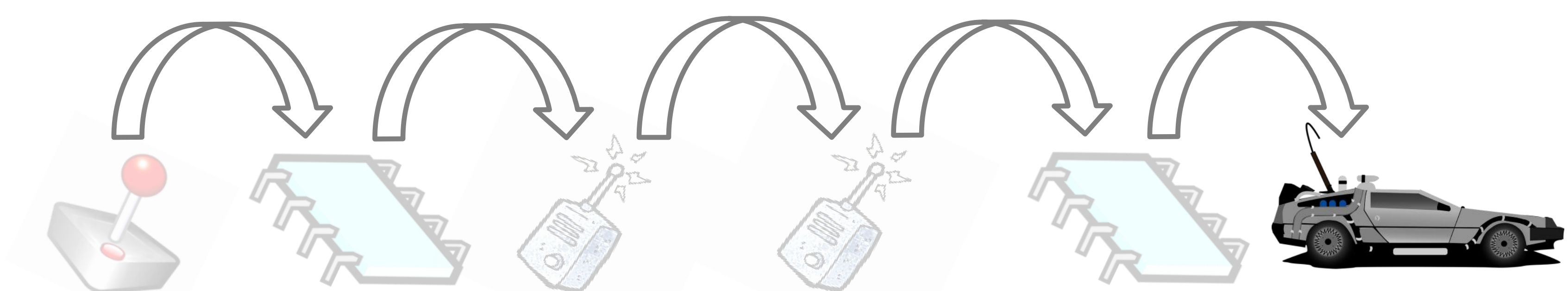


Receiver - Motor Controller

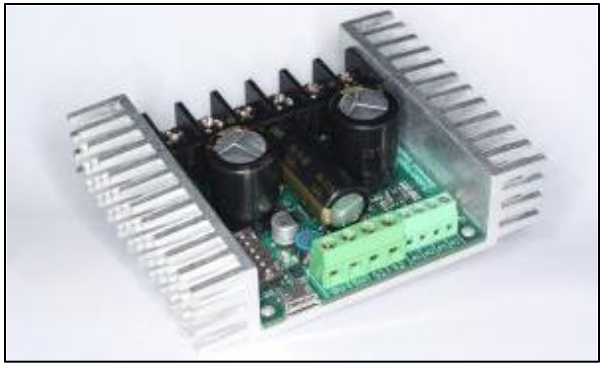


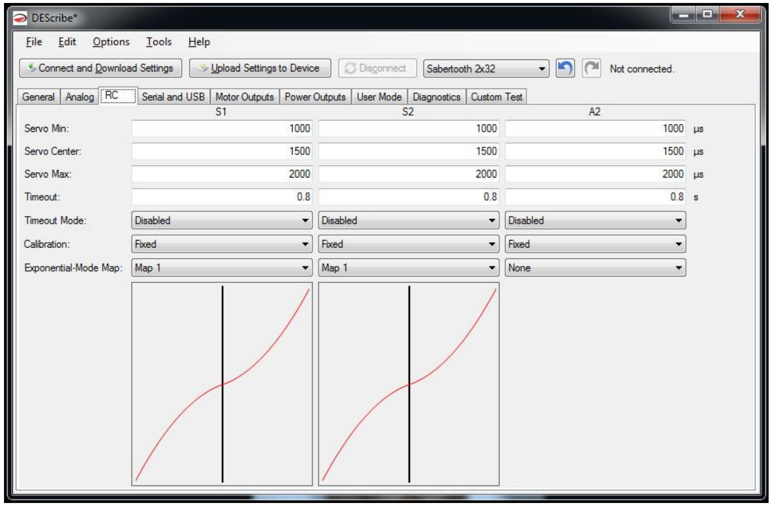
Dimension Engineering Sabertooth 2x32

- Regenerative drive uses excess mechanical motion to recharge batteries
  - Fast braking and longer battery life
- Additional accessory power outputs can be used to control electromagnetic brakes, optional accessories, or as voltage clamps
- Handles 6 – 30 V
  - 32A continuous (64A peak)
- Rated for combat robots up to 100 lbs or hobby robots up to 300 lbs
- Cost = $124.99



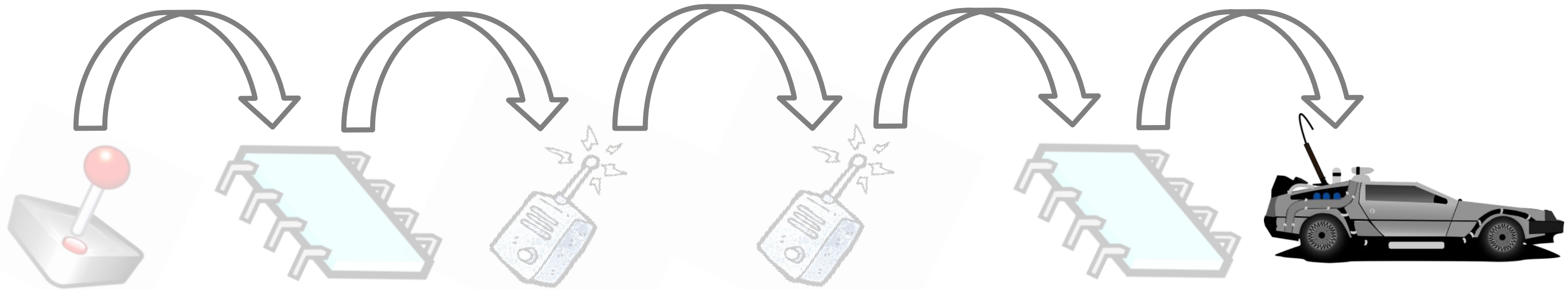
Receiver - Motor Controller



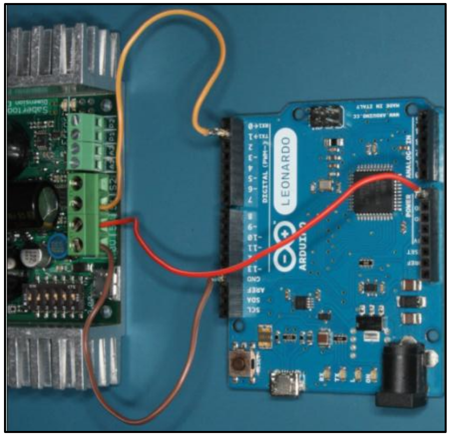


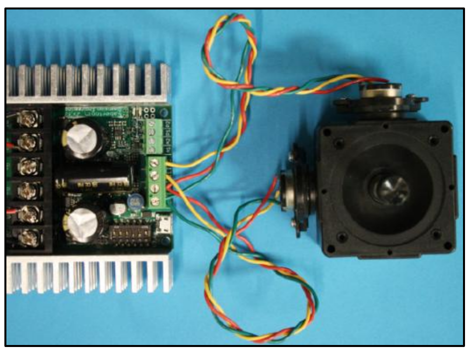
Dimension Engineering Sabertooth 2x32

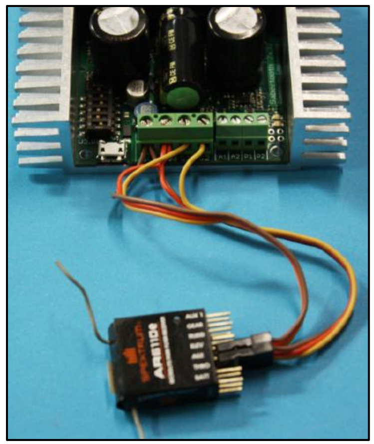
- 57 page user manual + additional downloadable documentation
- DIP switch settings allow for configuration in the field
  - Free online Wizard to simplify the process
- USB-configurable current limiting, control, monitoring, and serial output options; allows for power curve adjustments in a graphical user interface via free software



Receiver - Motor Controller

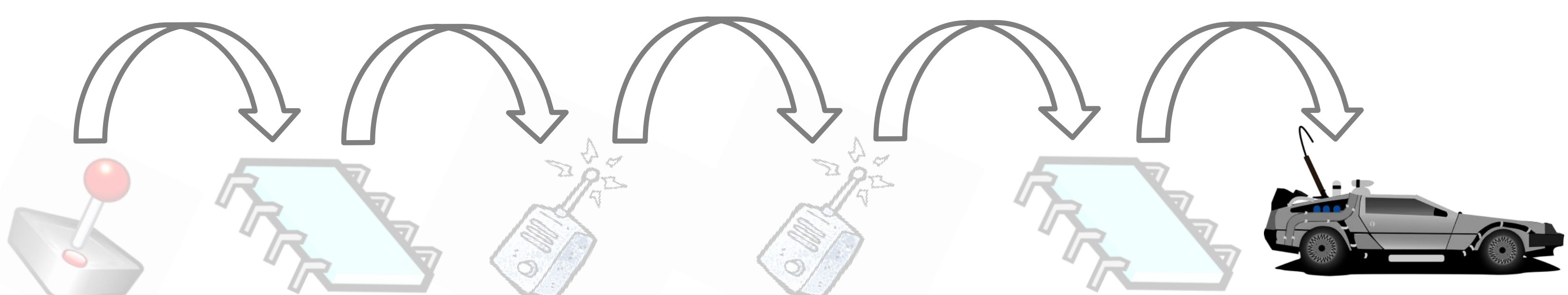




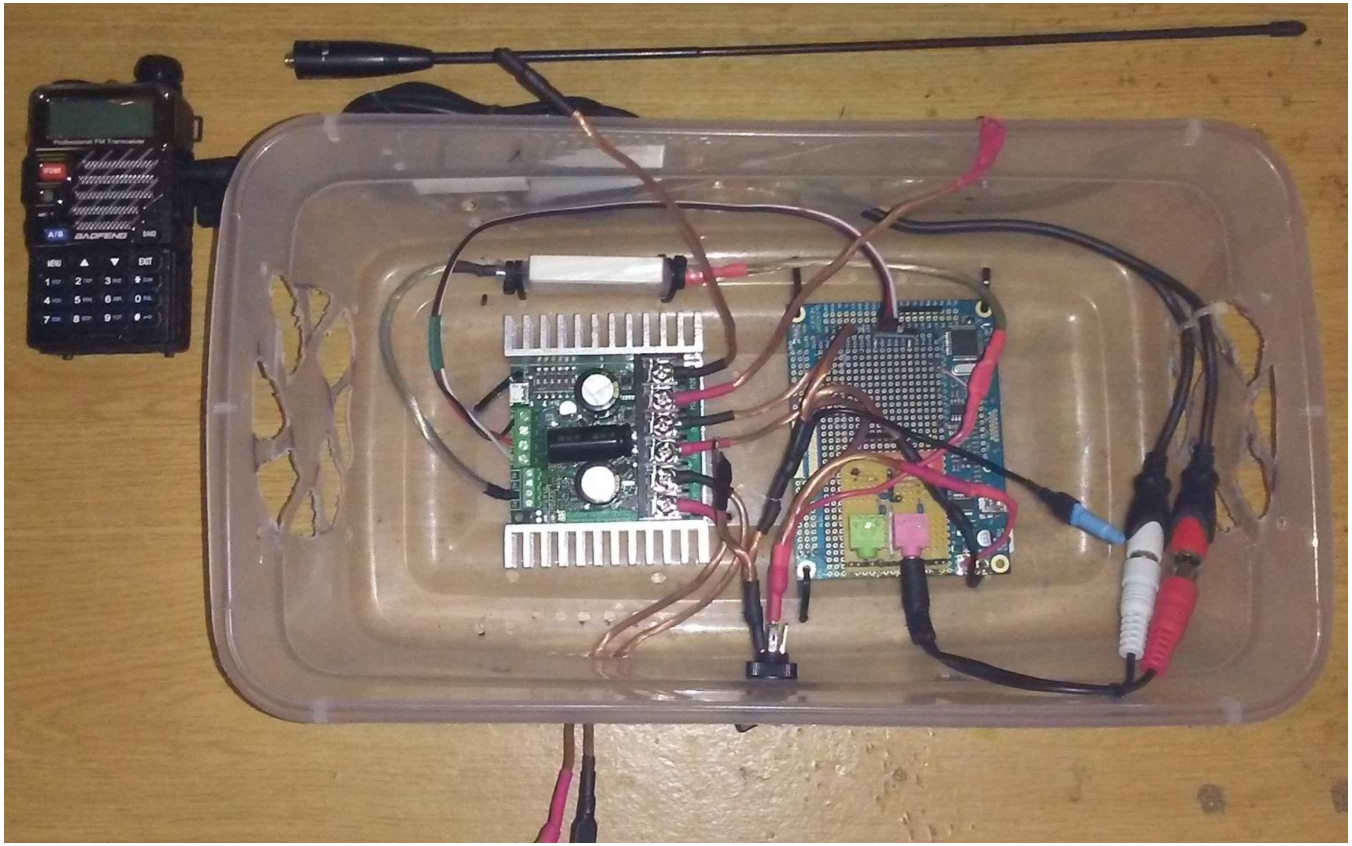


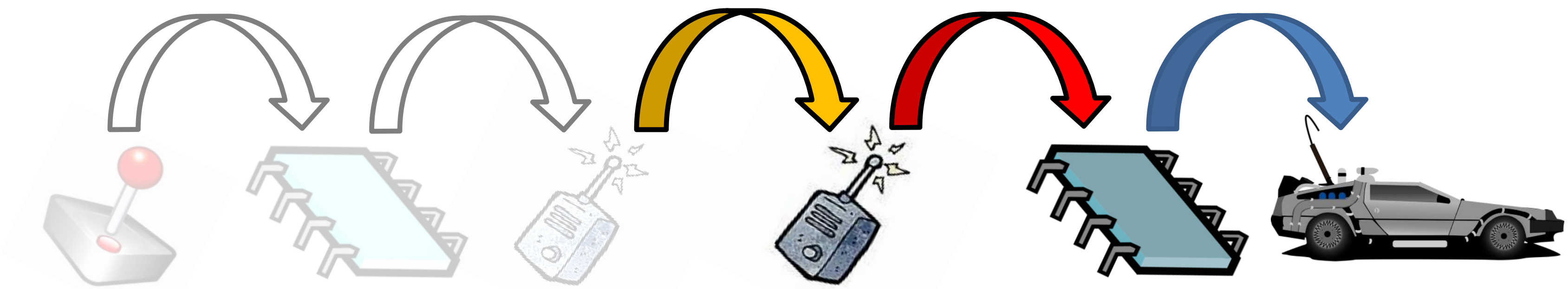
Dimension Engineering Sabertooth 2x32

- Input modes include:
  - Analog, RC servo pulses
    - compatible with a wide array of transmitters :
      - RC hobby transmitters
      - Microcontrollers
  - USB serial (plain text & packetized)
    - Compatible with:
      - PCs
      - Single-board computers



Assembled Receiver

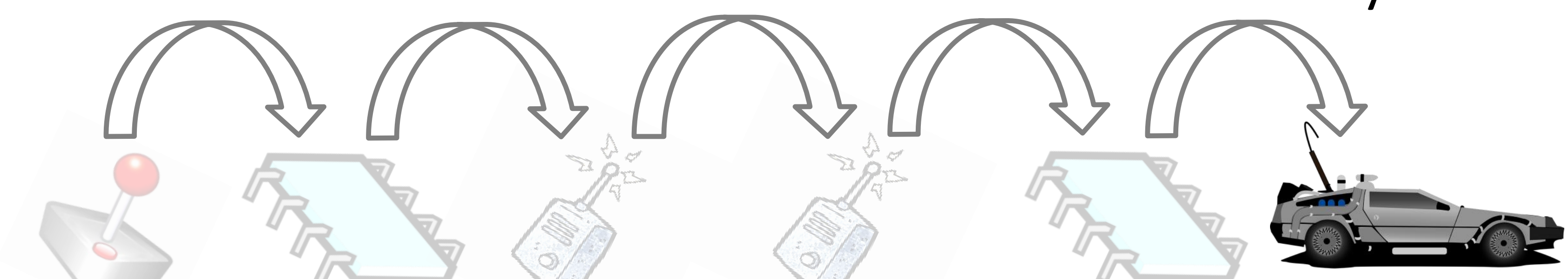




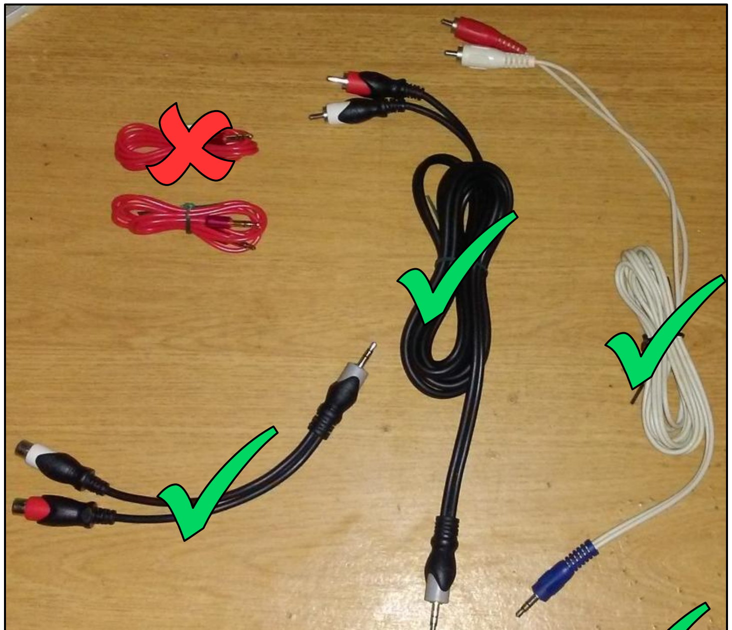
Vehicle

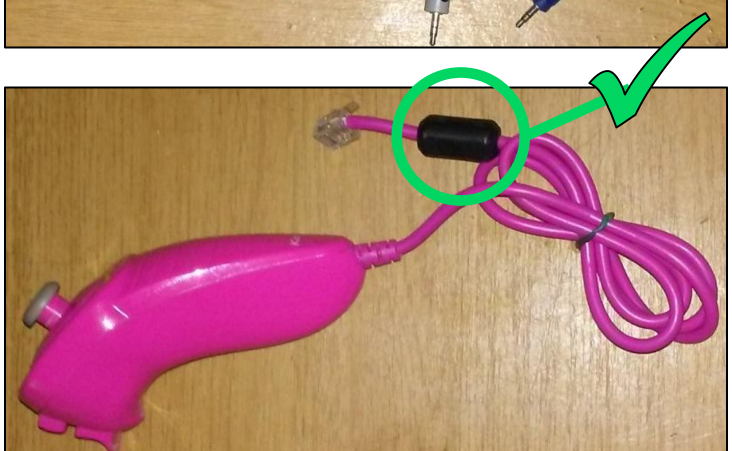
Powerwheels Cadillac Escalade Barbie Edition:

- Weight limit = ~ 130 lbs
- Stripped down and painted by student volunteers
- Modified for differential drive
  - Removal of front wheels, axle, & steering column
  - Replaced with inflatable caster wheel
- Powered by 12V lead-acid lawn-mower battery



Lessons Learned – RFI & Noise Mitigation





Audio Cables

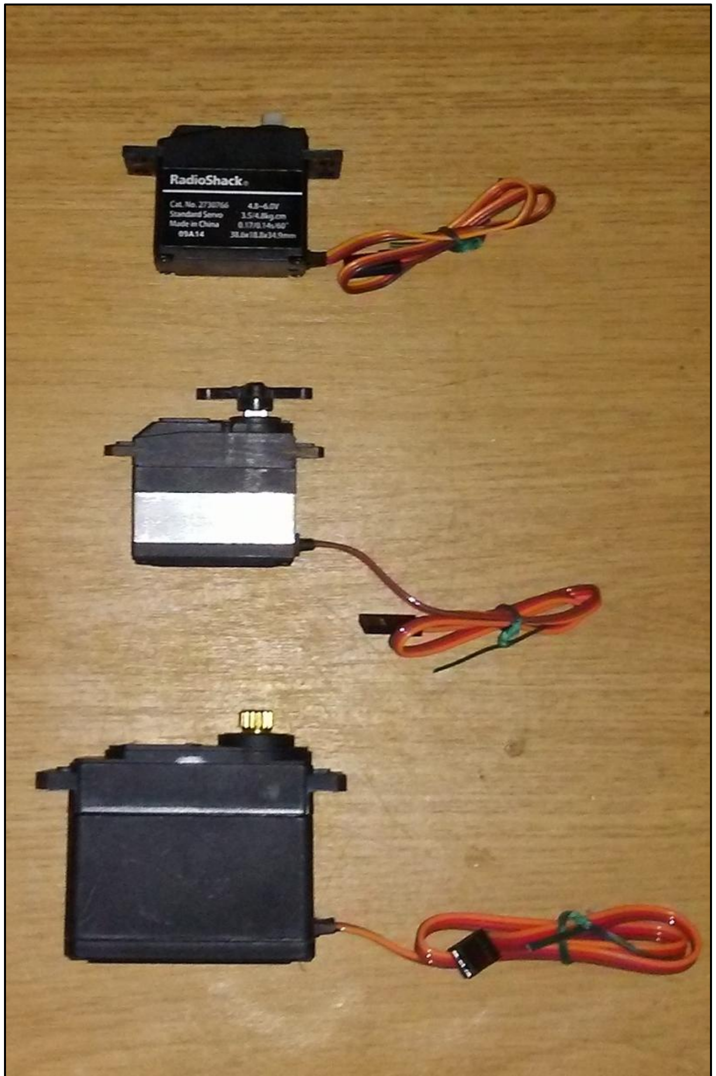
- Use high-quality, shielded cables
  - Cheap cables can serve as a source of interference & loss
- Check functionality of cables thoroughly & regularly
  - RC vehicles such as this can subject them to increased wear & tear
- Secure cables & connectors to prevent accidental disconnection

Data Cables

- RFI and induced interference from motors can corrupt the I^2C data-stream
  - Ferrite chokes actually work!

Checksum for FEC / Rejection

Lessons Learned – Steering



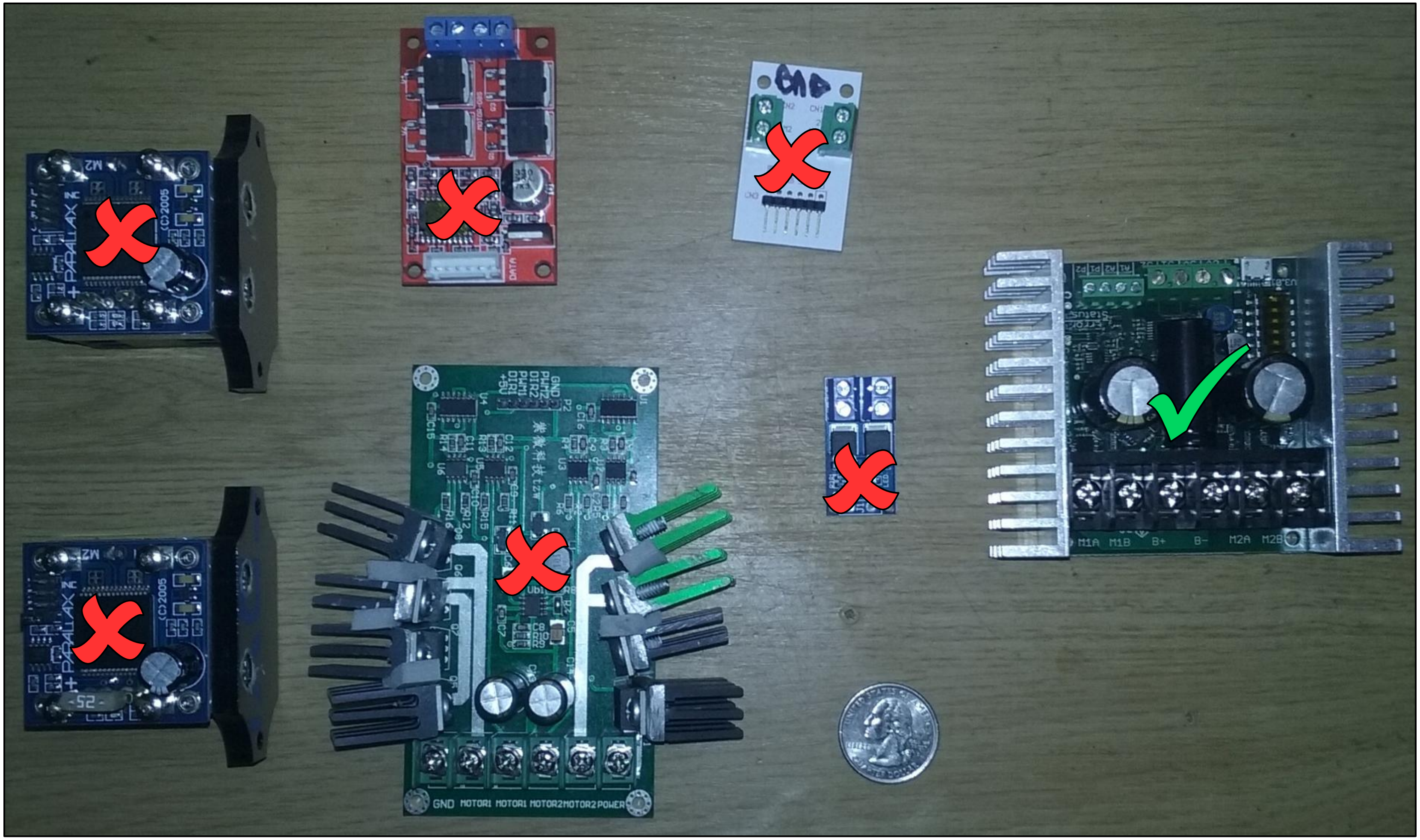
Original Plan

- Use servos, steppers, DC motors with encoders to steer the front wheels by rotating the steering column
  - Requires ↑ torque
- Would allow for both rear wheels to be driven in the same direction by a single (high-current-capacity) motor controller
  - Easier said than done!

Revised Plan

- Replace front wheels and steering assembly with a passive third wheel
- Allows for:
  - Differential drive
  - Zero-turning-radius
  - Regenerative braking

Lessons Learned – Motor Controllers



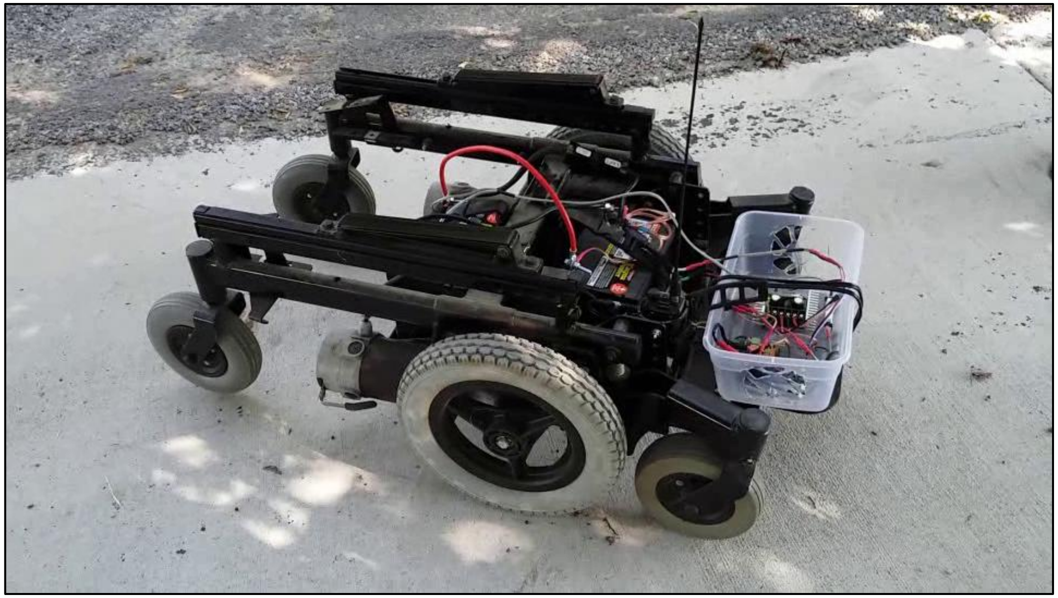
When it comes to motor controllers, you get what you pay for!

Lessons Learned – Vehicle Weight & Traction

A less-than-elegant solution!

Lessons Learned – Vehicle Weight & Traction

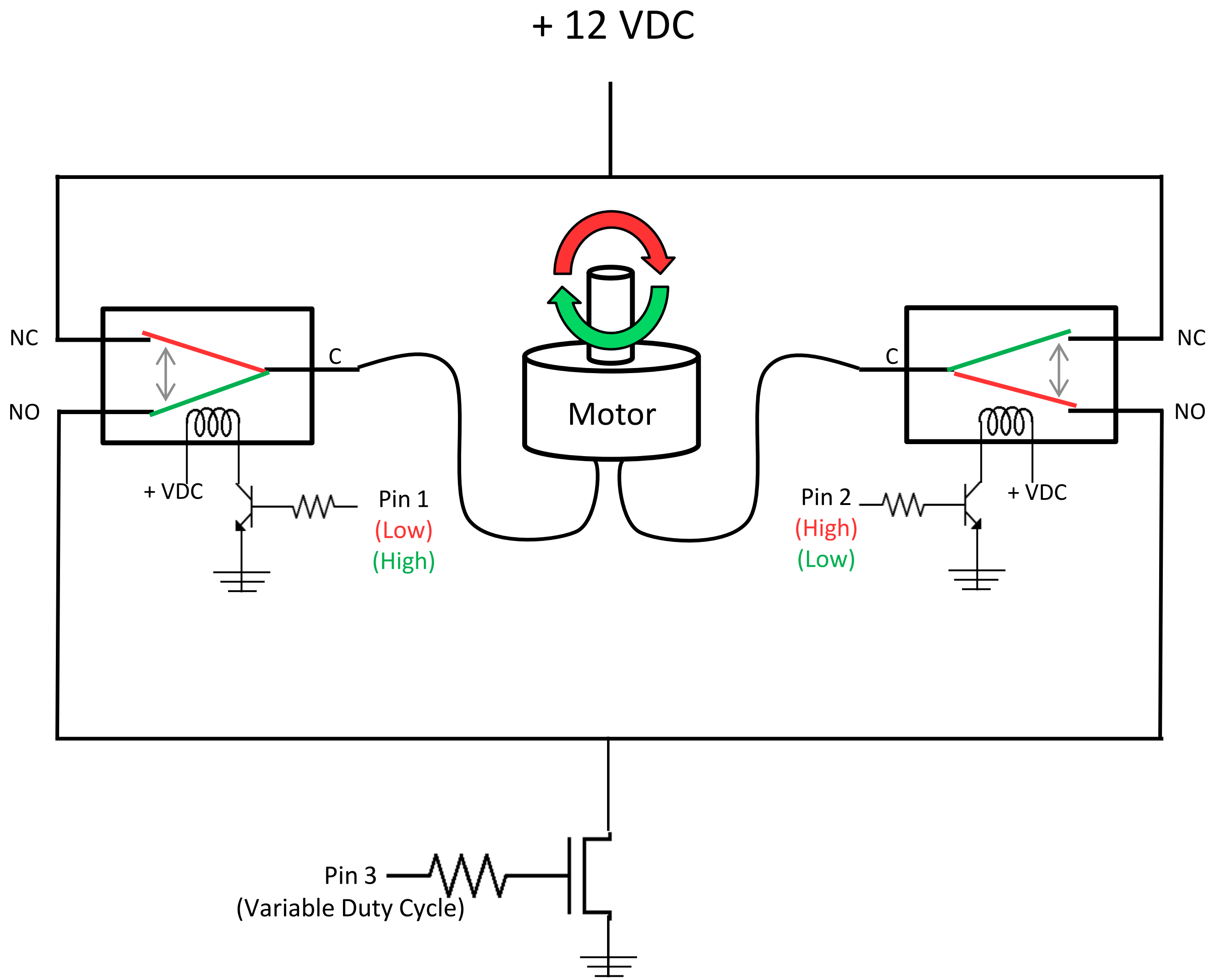
Future Directions – More Power!



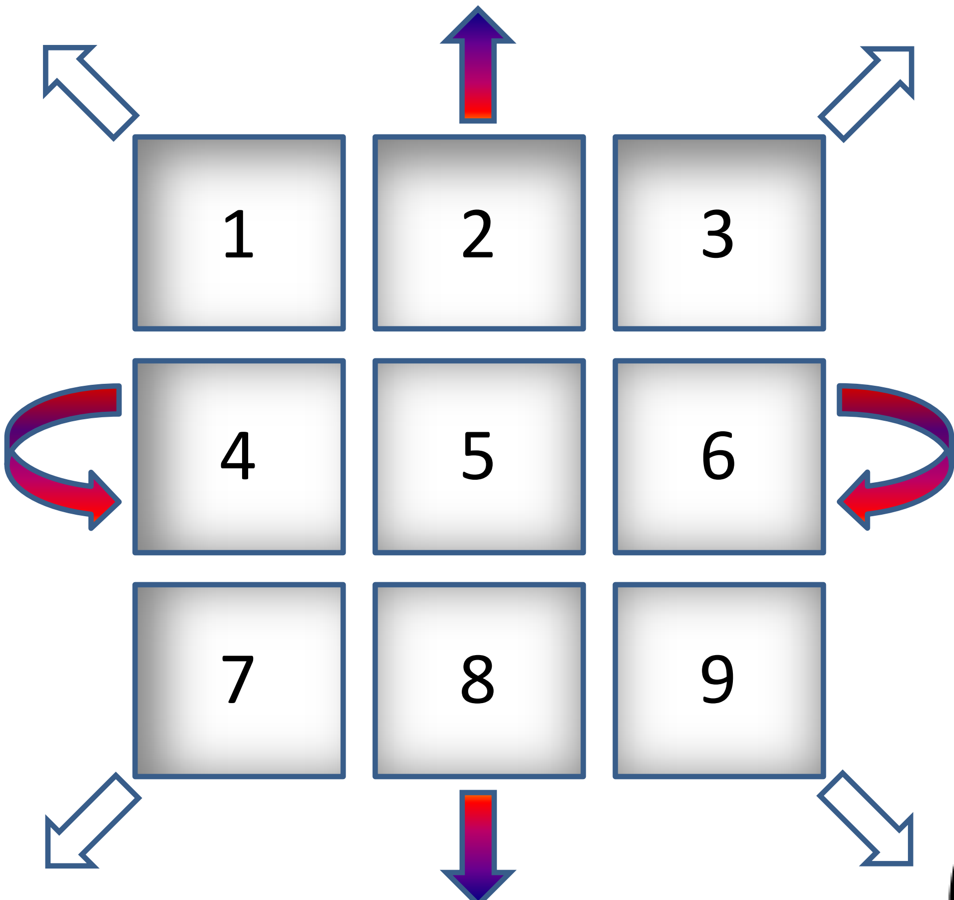
Future Directions – Smaller Platform

- Less weight
- Less work for the motors
- Less power handled by motor controller
- Uses a less expensive motor controller

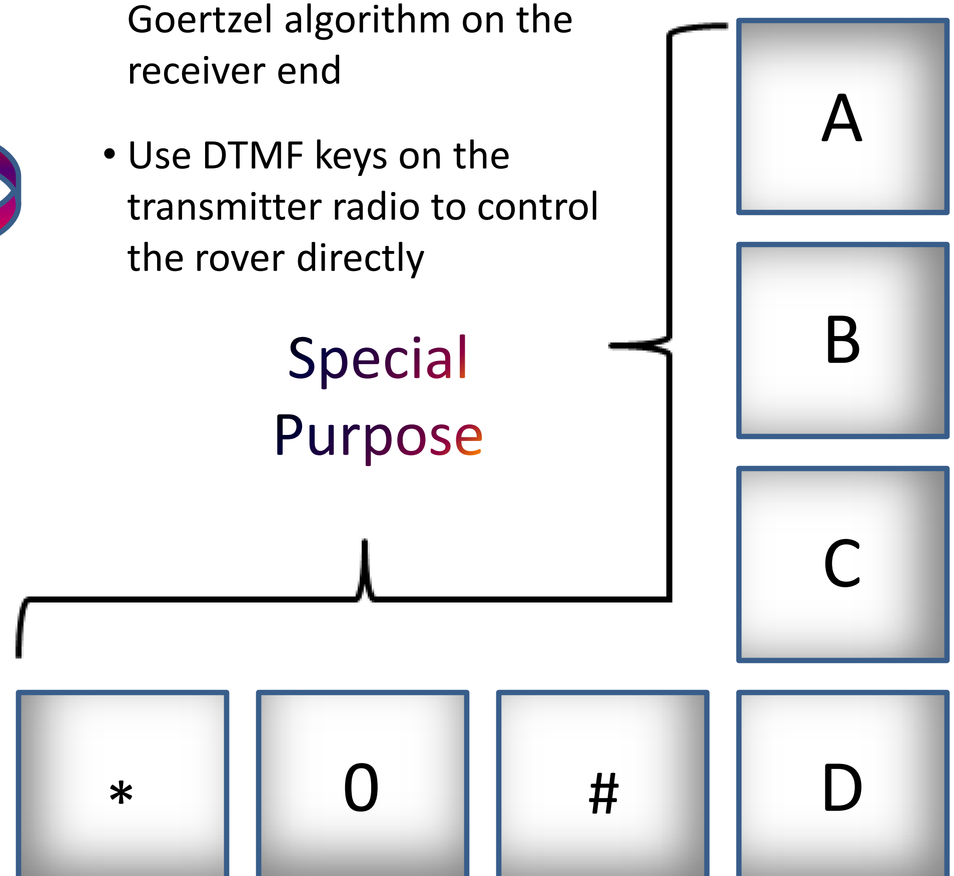
Future Directions – DIY Motor Controller



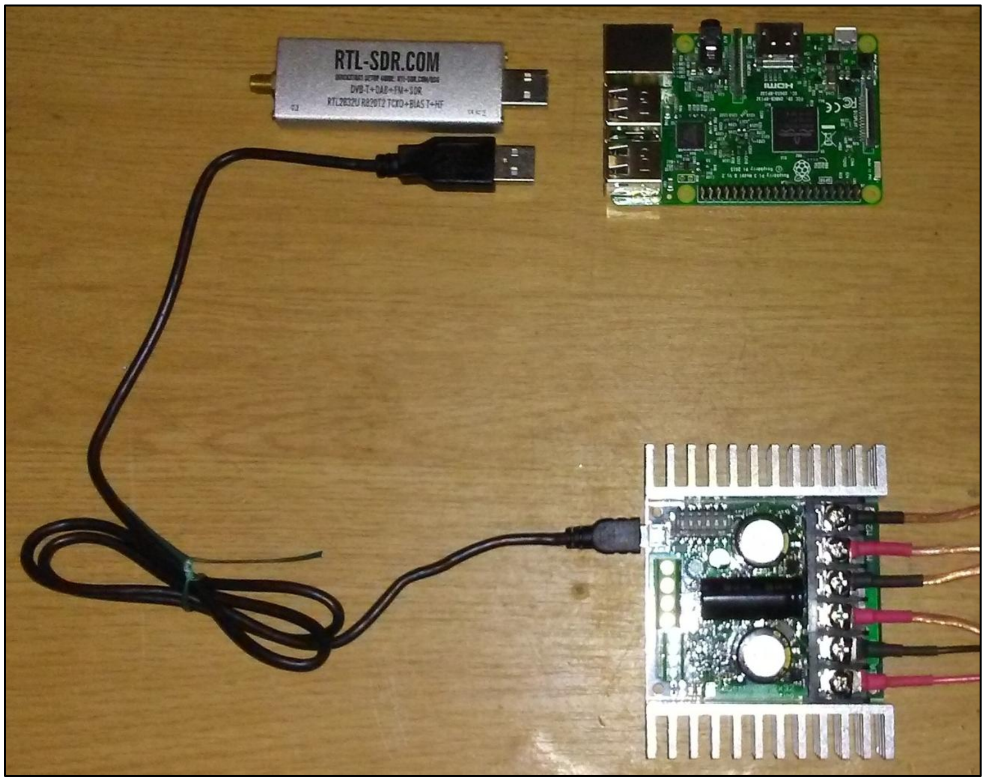
Future Directions – DTMF Control



- Eliminate the need for a modem and Nunchuck controller on the transmitter end by implementing the PhiPi Goertzel algorithm on the receiver end
- Use DTMF keys on the transmitter radio to control the rover directly



Future Ideas – Raspberry Pi



Raspberry Pi (~ $35 US)

- A linux-based single-board computer with open-source SDR and packet software

RTL-SDR USB Dongle (~ $15 US)

- A DVB-TV receiver hacked to function as an SDR receiver

Goal

- Use this system in place of HT and modem
- Implement APRS-like control system (unconnected protocol)
- Telemetry back-channel

Questions?