Workshop

Building multirotors
for fun
and frustration

Justamultirotorworkshop

(aka: it's not too late to run)

What you will learn

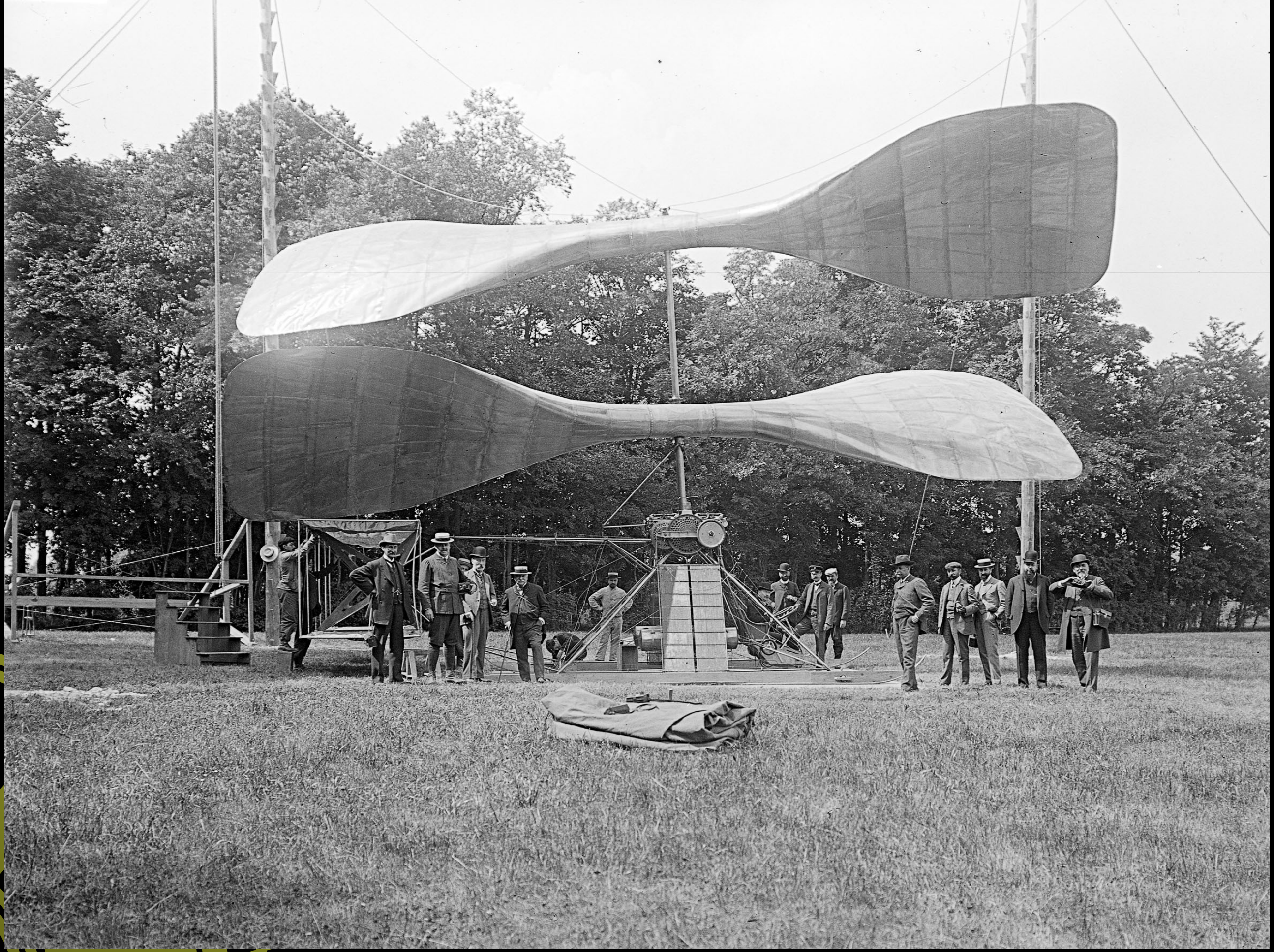
- What they are
- How they work
- How they're put together
- What it takes to design one
- How to put one together
- How long justa can make his presentations

Introduction

History of Multicopters

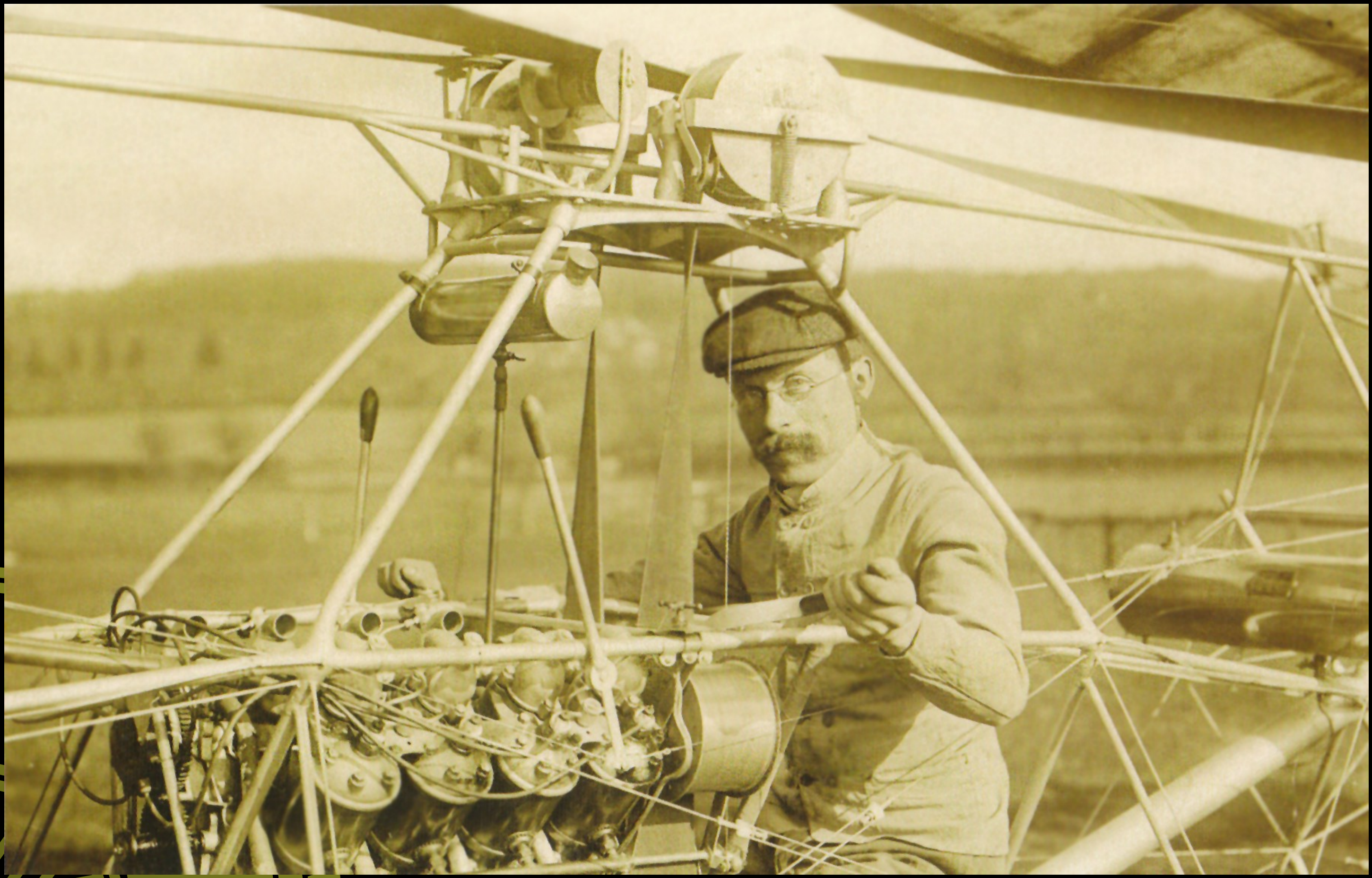
mul·ti·copter, (noun) \’ml-t-käp-tr’\

First Post



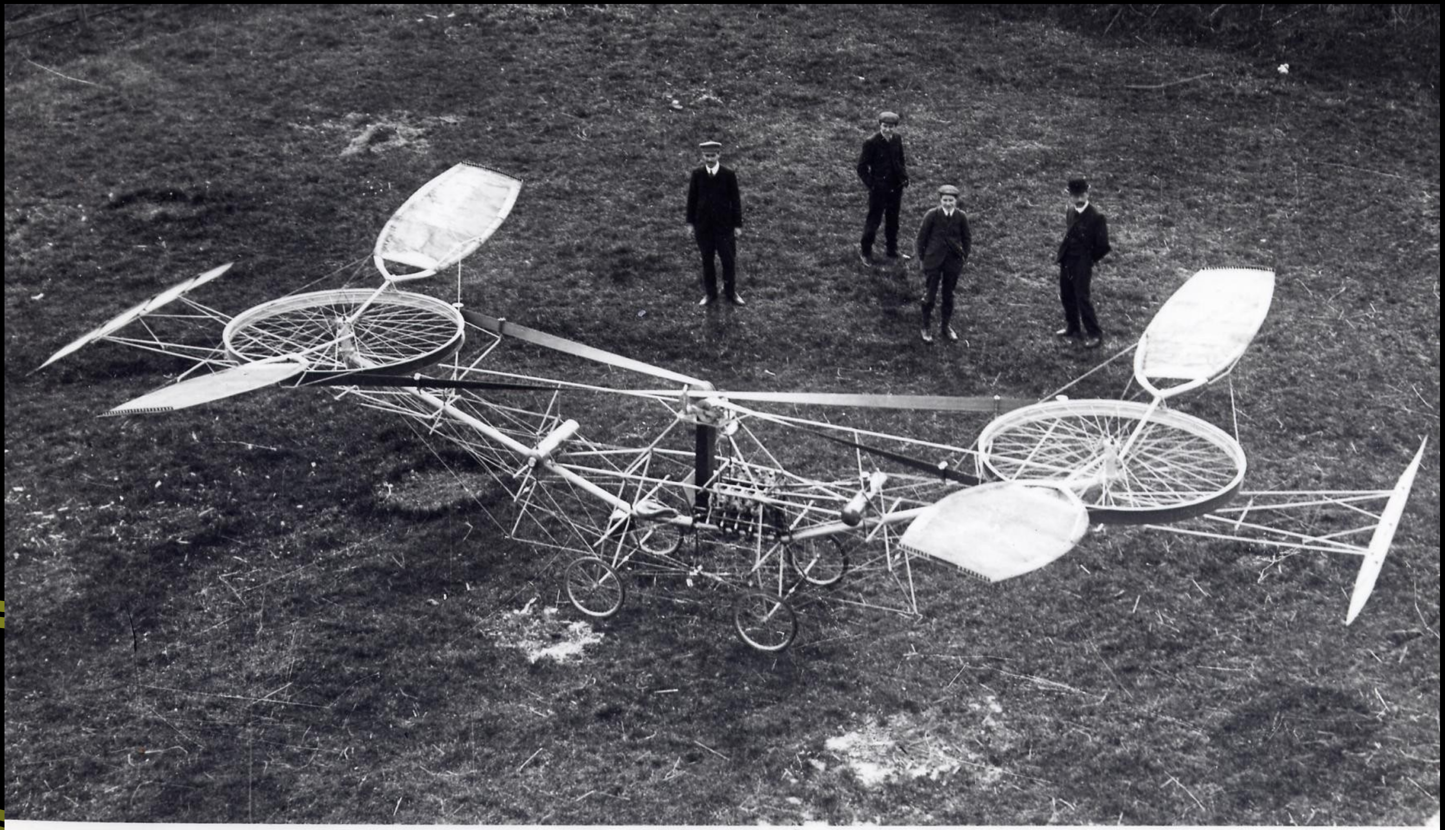
Maurice Leger, June 13, 1907

LOL, U FORGOT PERSON



Paul Cornu, 13 Nov, 1907

This can fly



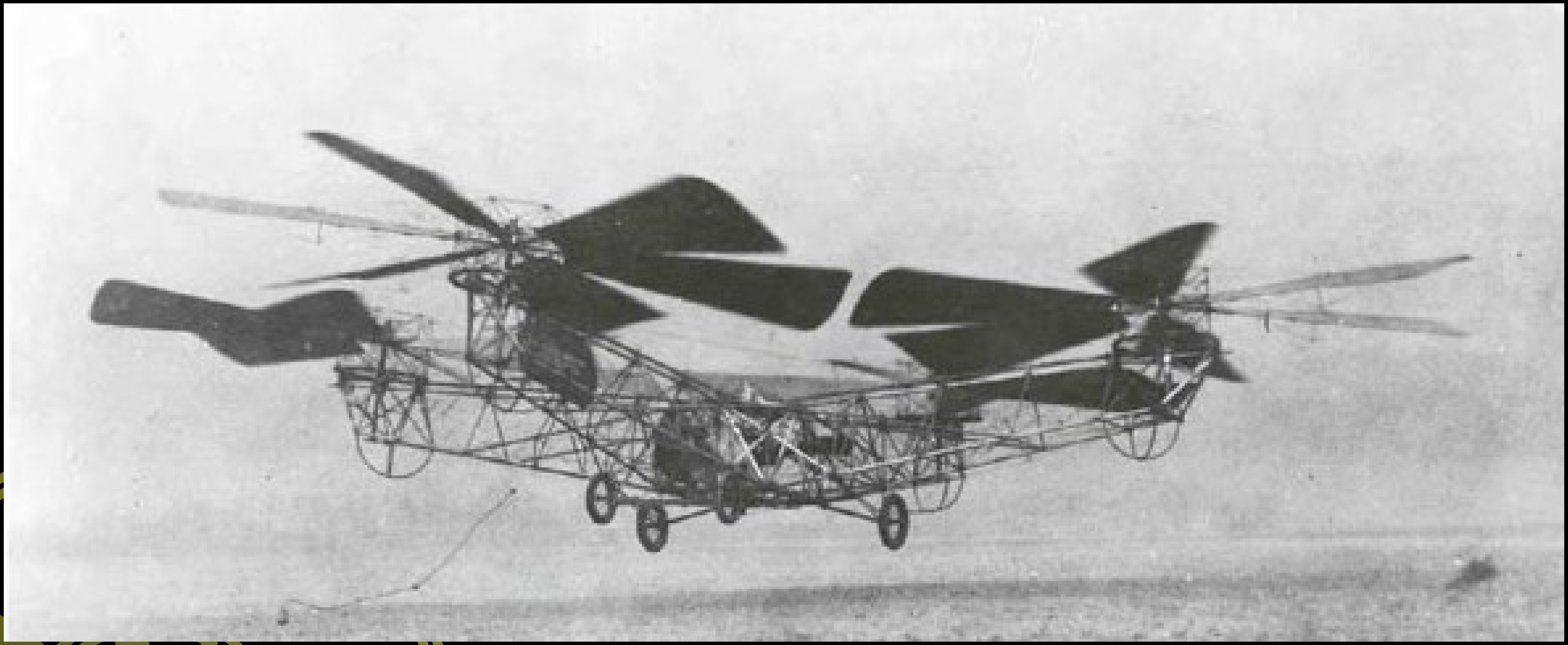
1907- Paul Cornu, à Lisieux, sur un appareil de sa conception réussit un soulèvement libre. (méthode de Charles Renard)

(for at least 20 seconds)

Disclaimer

Management hereby promises
flighttimes of over
20 seconds

Bothezat (wut)



Four is better, October 1922

mul·ti·copter, (noun) \’ml-t-käp-tr’\

- Has 2 or more rotors
- Mounted horizontal-ish
- Flies
-
-
- Profit!

Now: Swarms

Now: photography

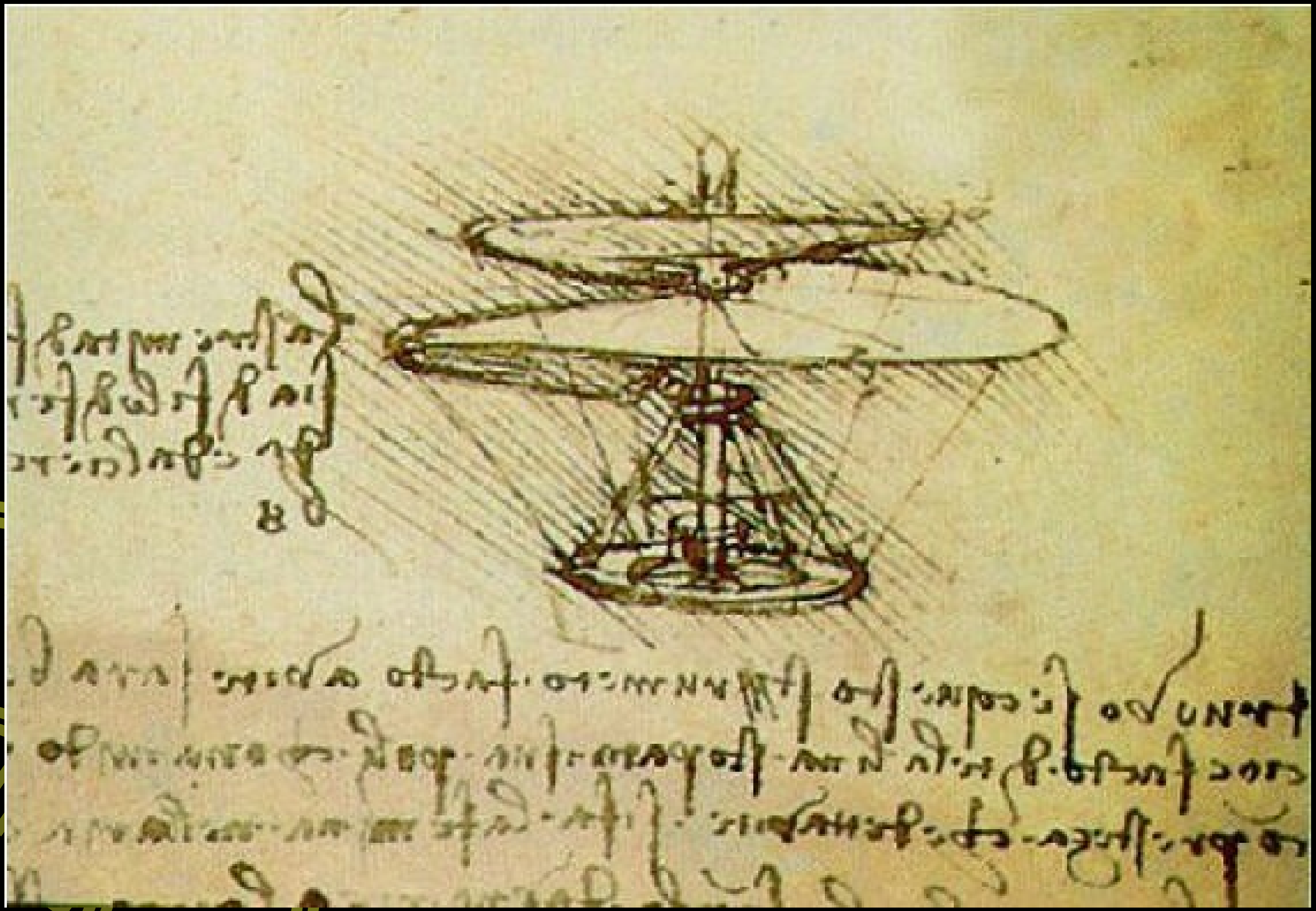
Now: Architecture

Now: porcine agriculture

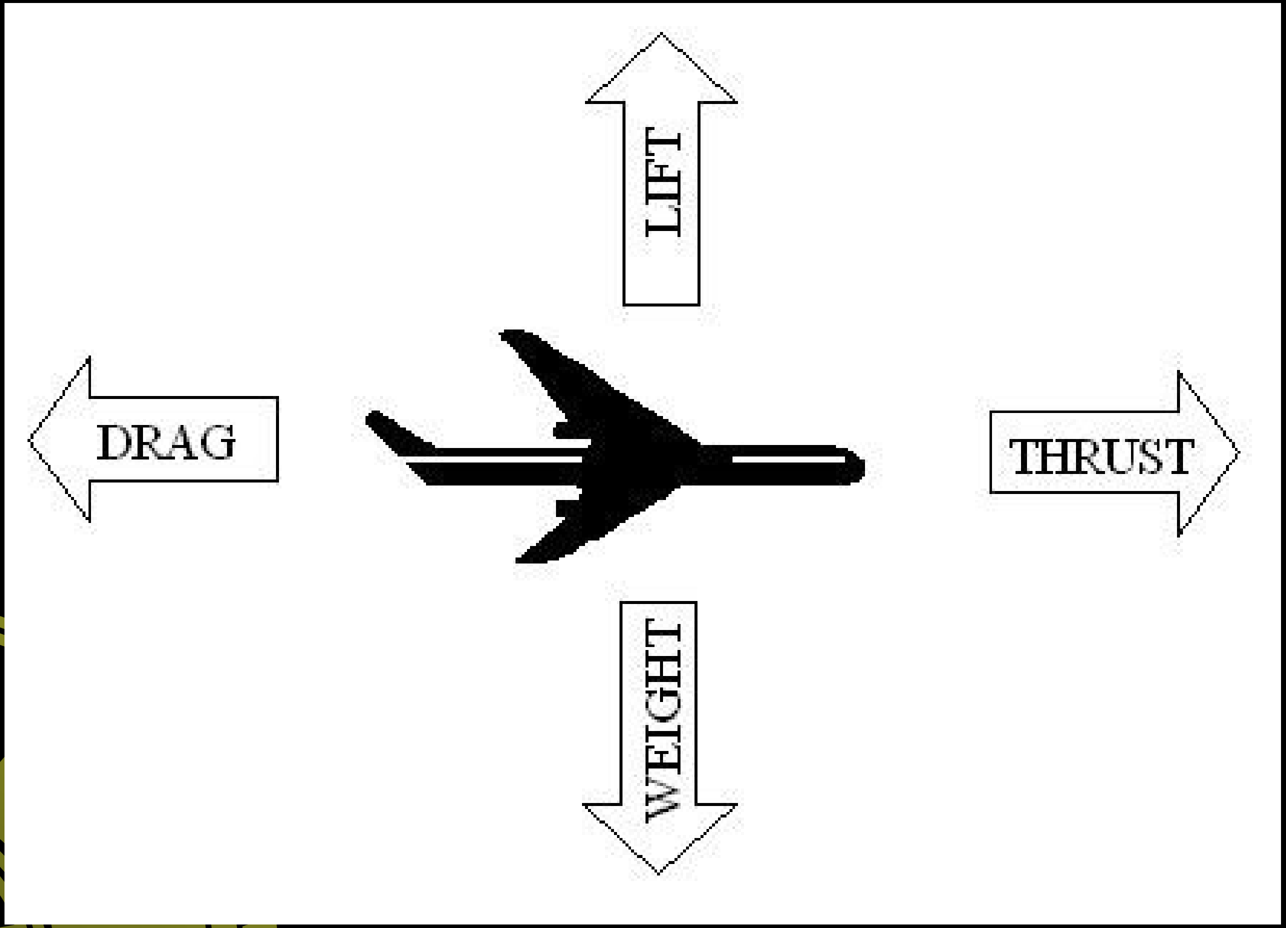
Now: DIY systems

How it flies

Principles of flight in just 101 slides

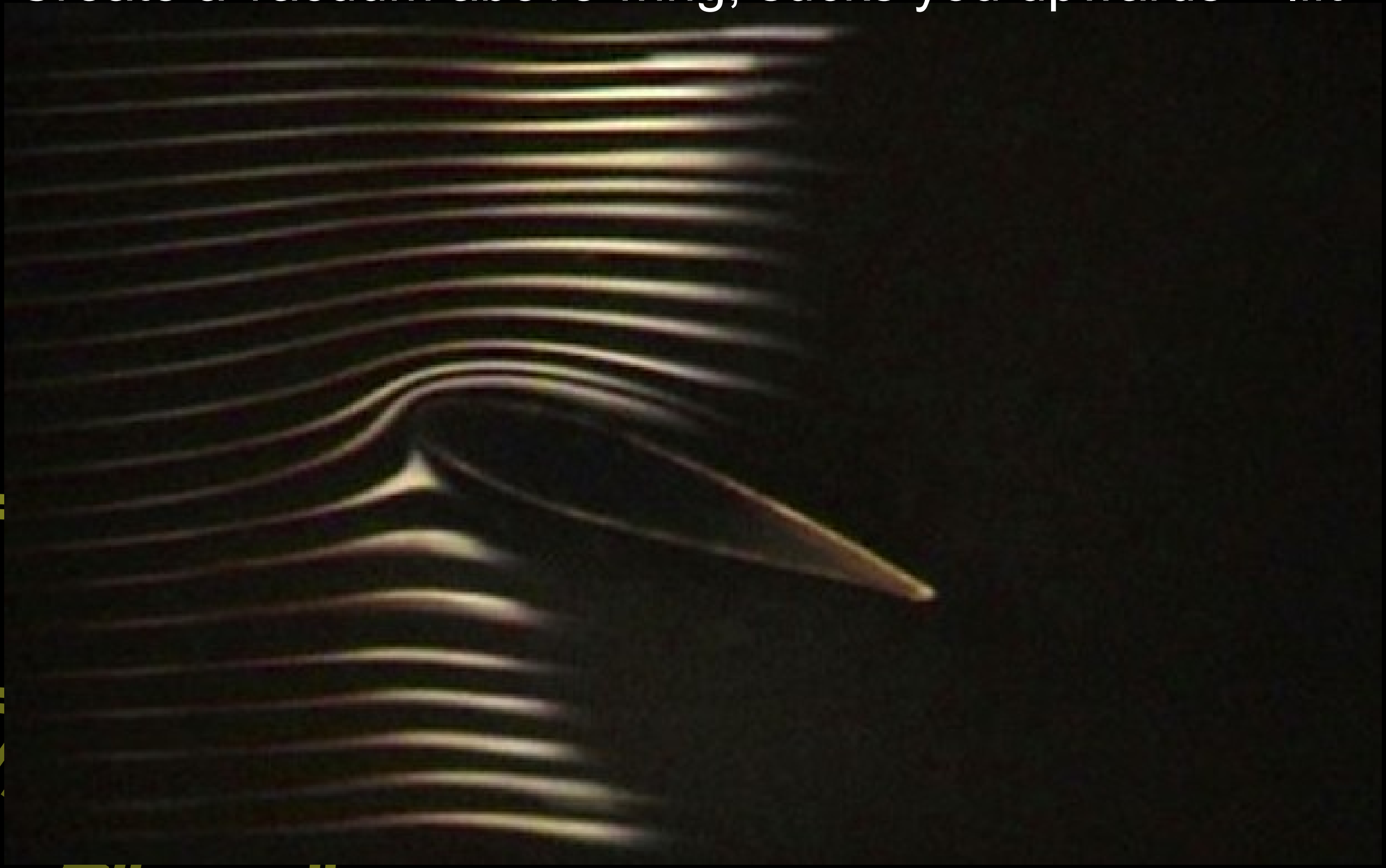


Thrust and Lift



Plane wings and rotors

Create a vacuum above wing, sucks you upwards = lift



Planes vs Rotor-craft

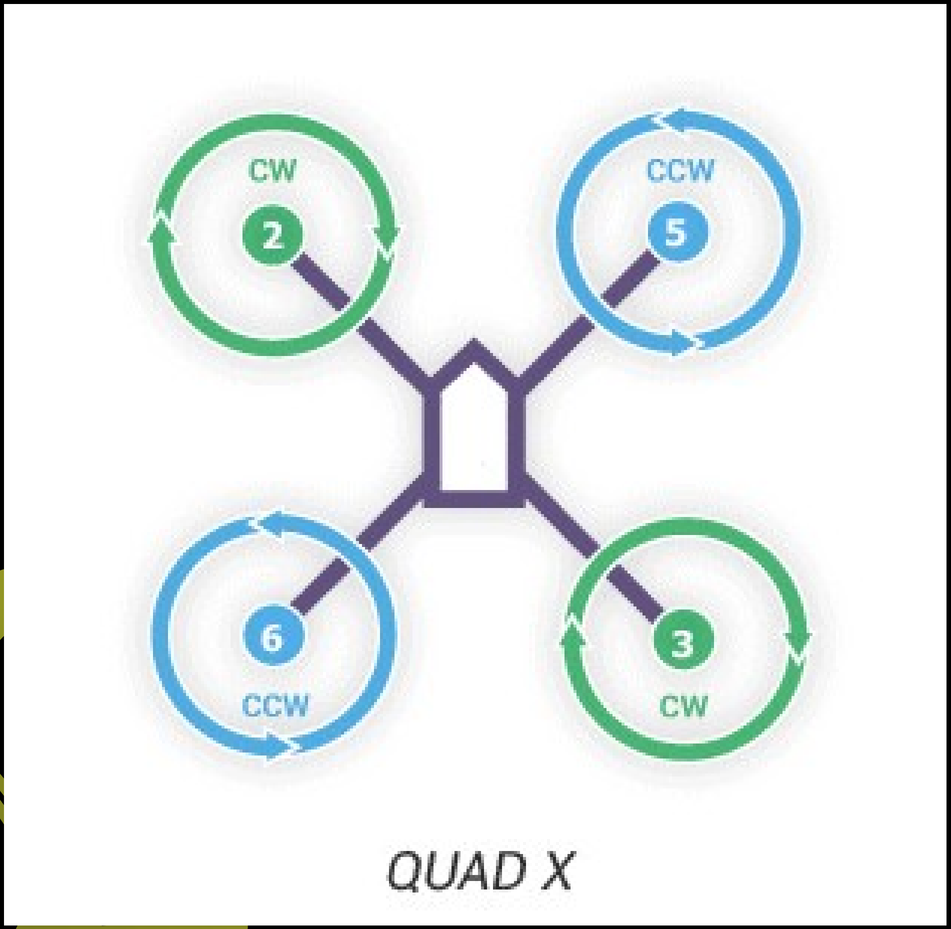
- Planes use rotors (or jets) for speed
- Speed is turned into lift by wings

- Rotorcraft use rotors directly for lift
- Trade-off: lose ability to glide

Multirotors vs Helicopter

- Both use rotors directly for lift
- Helicopters can use one rotor for lift
- Helicopters use small tail-rotor or jet to prevent spinning
- Helicopters use rotor-tilt to direct movement
- Helicopters often use rotor-PITCH for lift/speed control.

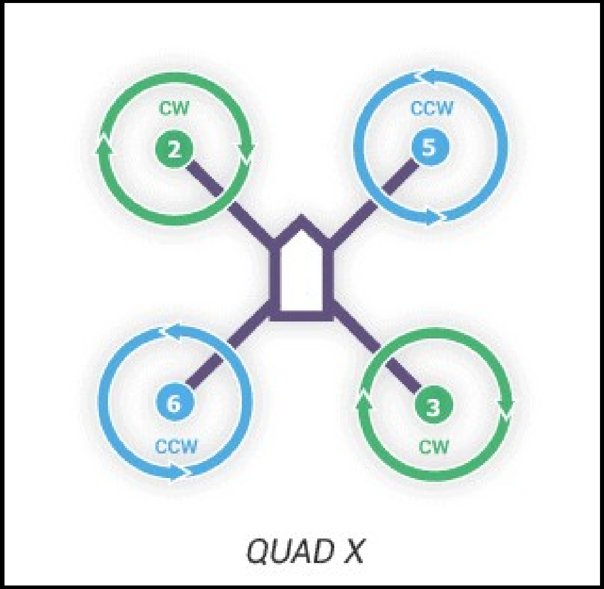
Multirotors



(Example; your quadcopter will look different from this picture)

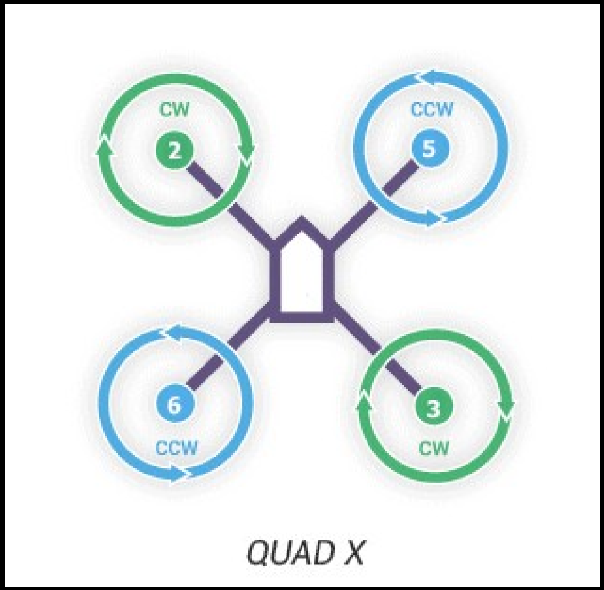
Going up! (or down, or staying)

- Keep lift on all rotors the same
- Speed 'm up for up
- Slow 'm down for down



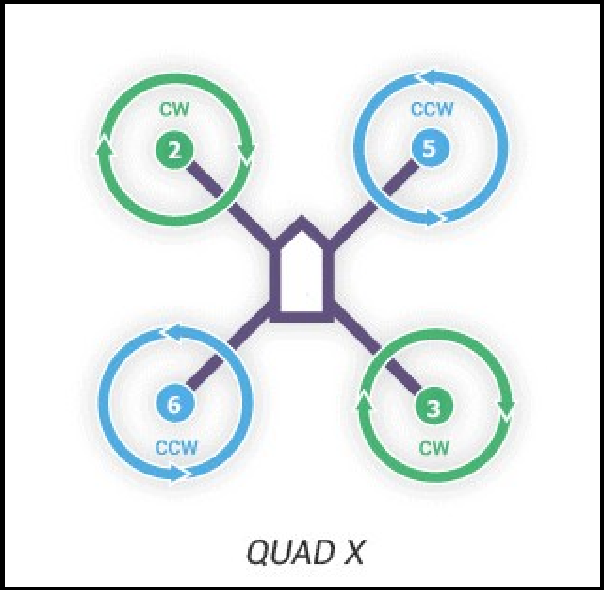
Directions

- Sideways or back/forth:
  - Throttle down rotors on one side
  - Throttle up rotors on opposite side
  - Aircraft will tilt
  - Thrust to 'highest' side
  - Movement to 'lowest' side



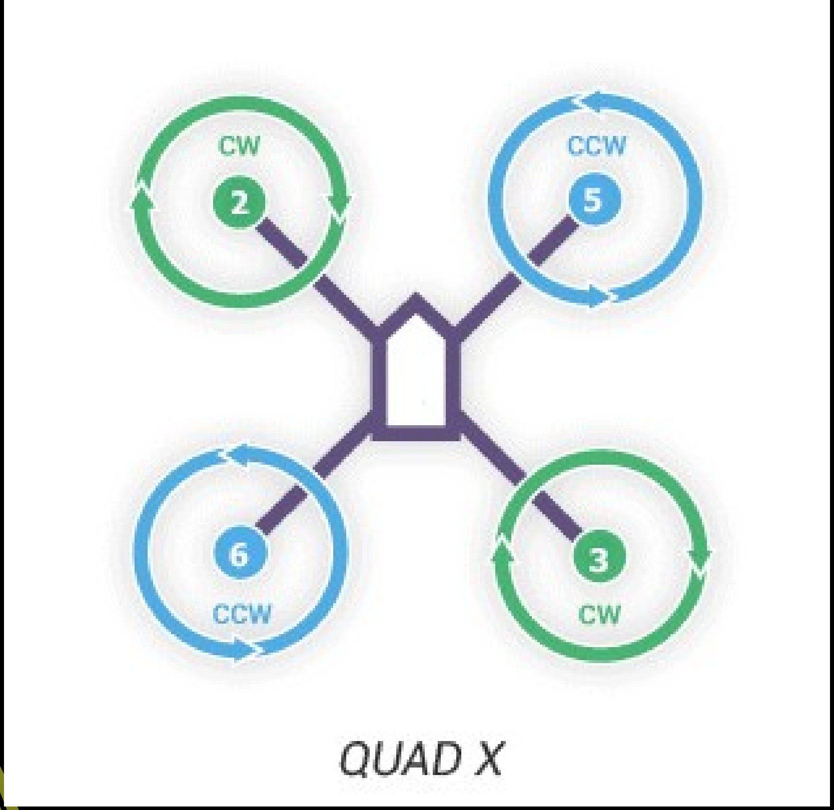
Turning

- Turning around your own axis
  - Wait..
  - No 'tail rotor'
  - Y U NO CRASH ALWAYS ?



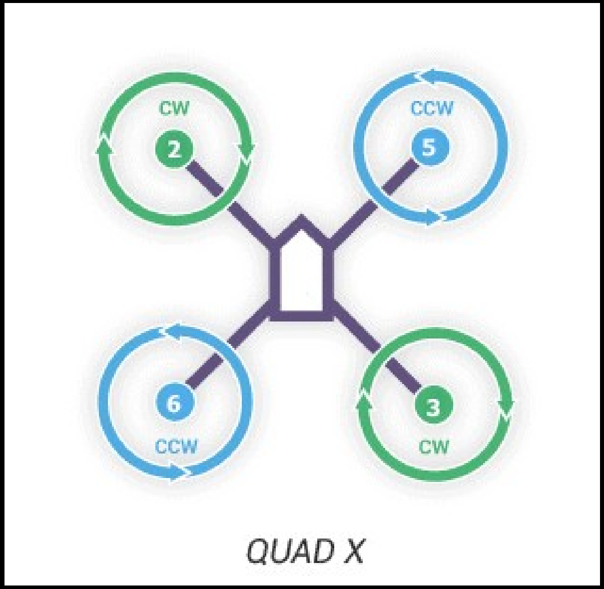
NOT Turning

- CW
- CCW

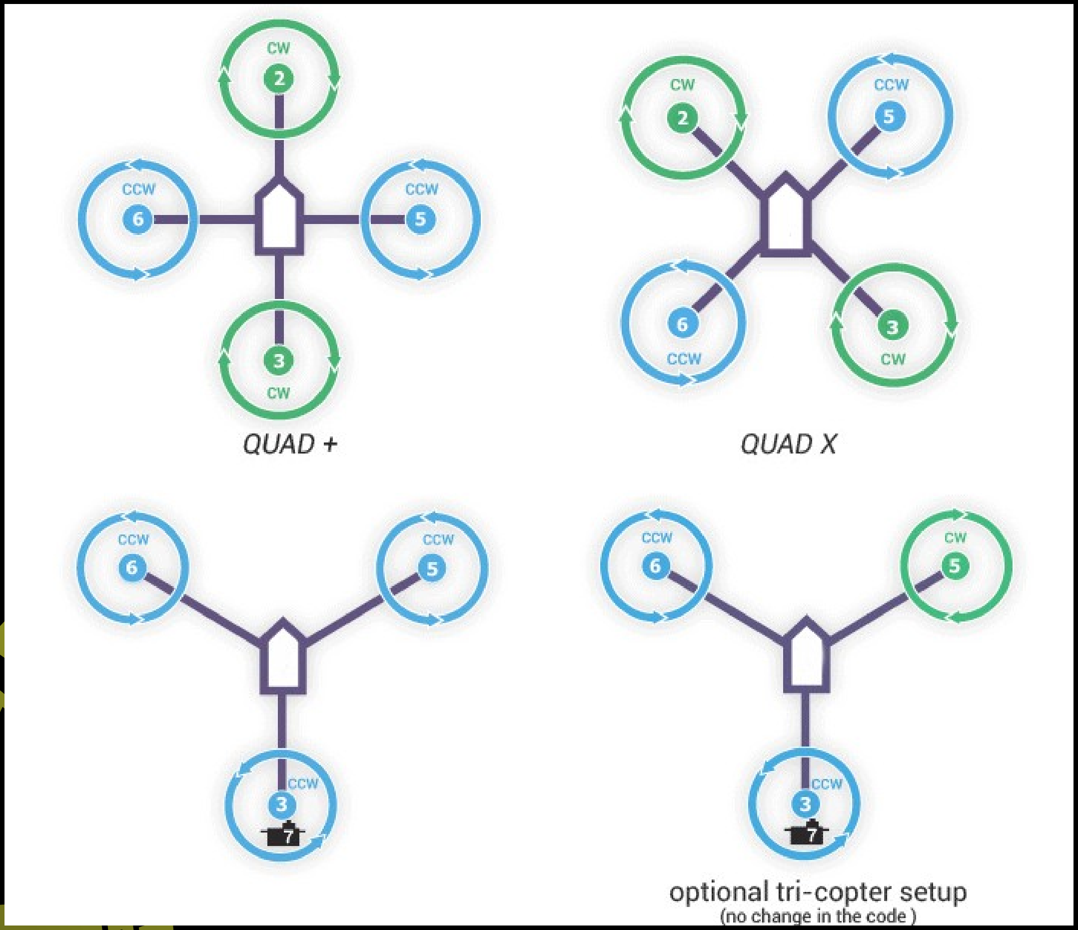


Turning (revisited)

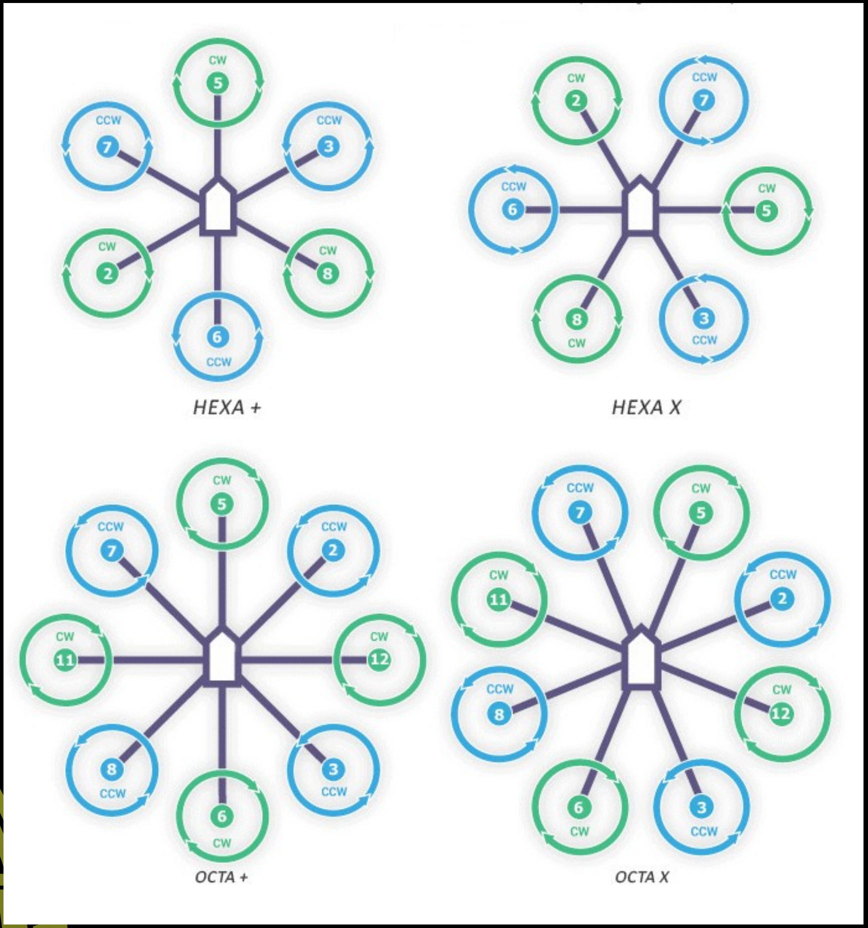
- Turning CW
  - Throttle DOWN CW rotors
  - Throttle UP CCW rotors



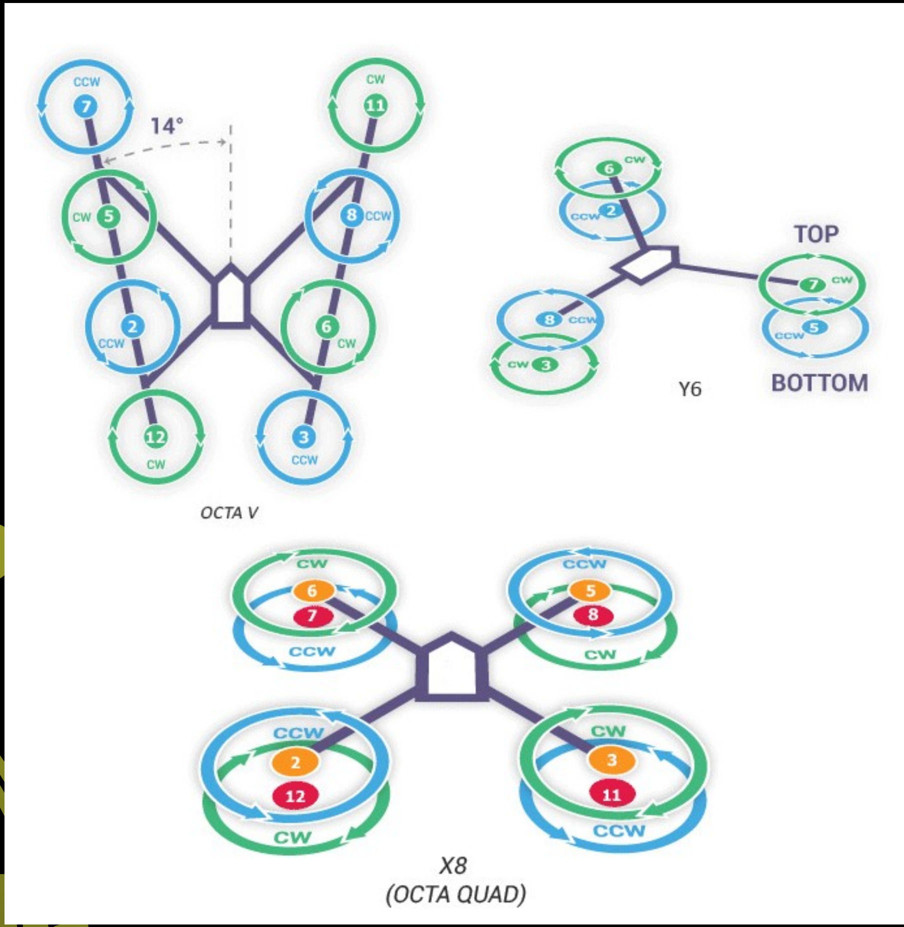
Same principle



These too


CW
5
CCW
7
CCW
3
CW
2
CW
8
6
CCW
HEXA +
CW
2
CCW
7
CCW
6
CW
5
8
CW
3
CCW
HEXA X
CW
5
CCW
7
CCW
2
CW
11
CW
12
8
CCW
3
CCW
6
CW
OCTA +
CCW
7
CW
5
CW
11
CCW
2
CCW
8
CW
12
6
CW
3
CCW
OCTA X

And these..



CCW
7
CW
11
14°
CW
5
8
CCW
2
CCW
6
CW
12
CW
3
CCW
OCTA V
6
CW
CCW
2
TOP
7
CW
CCW
5
8
CCW
CW
3
Y6
BOTTOM
CW
6
7
CCW
CCW
5
8
CW
CCW
2
12
CW
CW
3
11
CCW
X8
(OCTA QUAD)

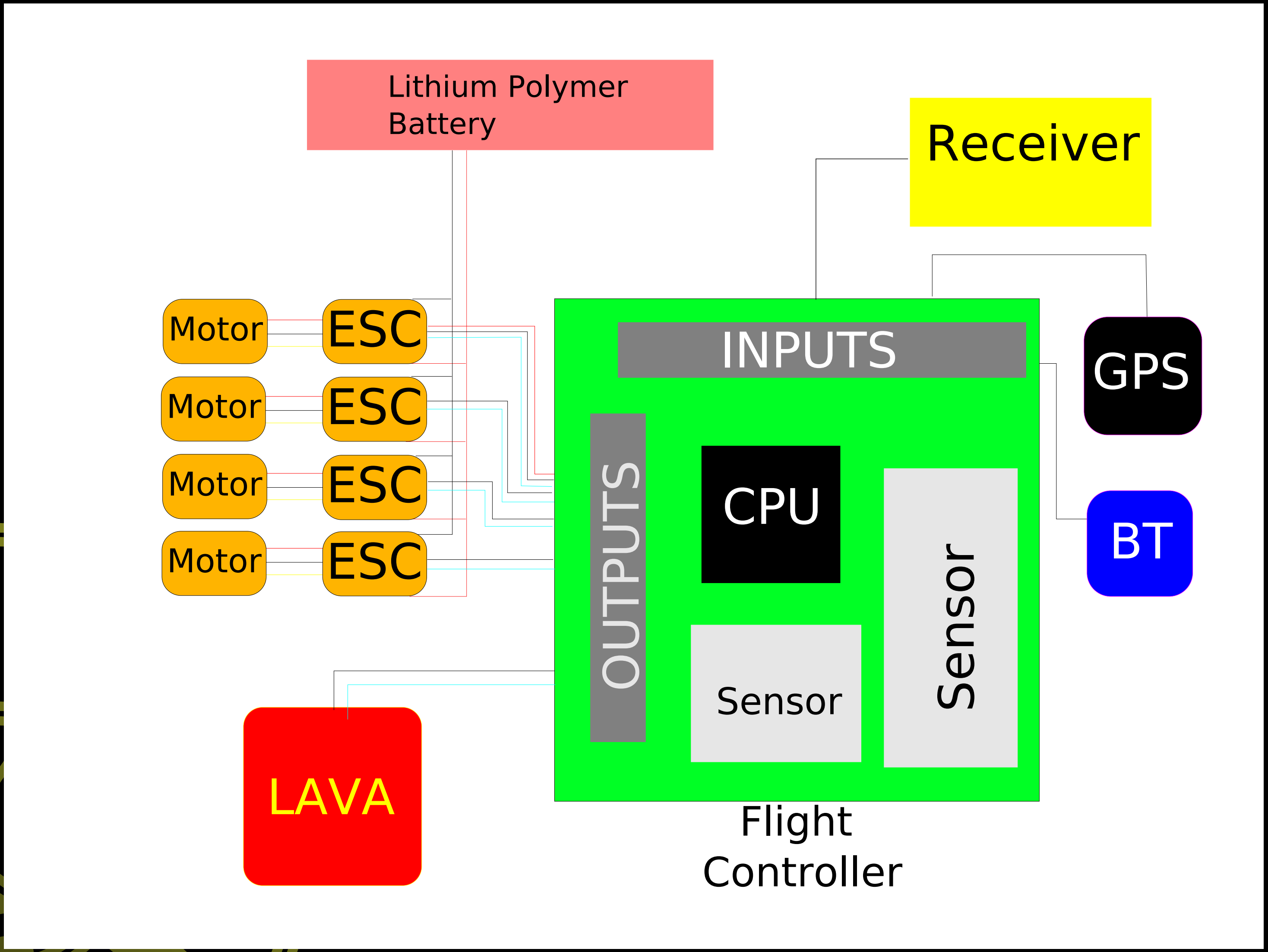
Possibly even this..

The nitty gritties

Global System Overview

- Transmitter/Receiver (TX/RX)
- Flightcontroller
- Motors
- Propellors
- Batteries (and power-distribution)
- (other stuff: GPS, BT, OSD, FPB, TELEMETRY, LEDs, ROCKETS, BEER)

The Parts



Lithium Polymer Battery
Receiver
Motor
ESC
Motor
ESC
Motor
ESC
Motor
ESC
INPUTS
GPS
OUTPUTS
CPU
Sensor
BT
Sensor
LAVA
Flight Controller

RX/TX

Plane/Heli TX

Car/Boat TX

TX: General


TURNIGY
power systems
9X
2.4G 9 CHANNEL
GLID / AIRPLANE / HELICOPTER SYSTEM
9X 2.4G 9 CHANNEL

RX


FLY SKY
BAT
CH6
CH5
CH4
CH3
CH2
CH1
AFHDS
6 CHANNAL 2.4GHz DIGITAL
RECEIVER SYSTEM

RX/TX Specifics

- Different frequencies and technologies
- Analog
  - 27Mhz, crystal-based
  - 40Mhz, crystal-based
- 2.4Ghz, Digital. 'binding'
- LongRange (LR), varies

Analog

Of crystal magic
and analog voodoo

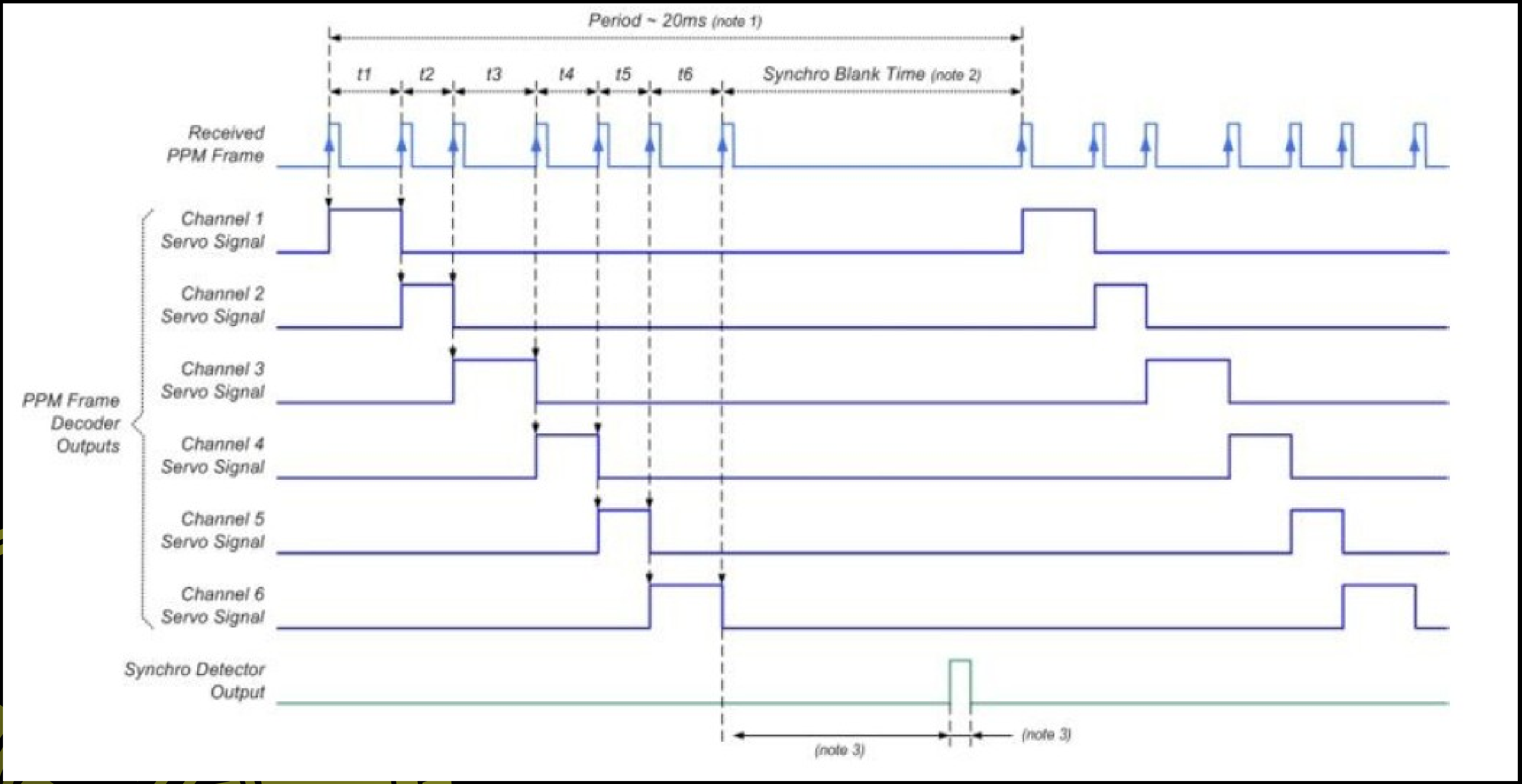
All is relative

- Problem: No absolute state
- Wish: Send absolute 'levels' or 'positions'
- Option: Amplitude ?
- Option: Frequency ?
- Option: signal interval ?
-
- Profit!

RX/TX terminology

- RX: Receiver
- TX: Transmitter
- PWM: Pulse Width Modulation
- PPM: Pulse Position Modulation
- PPMSUM: Combined 'raw' signal (more later)

Air->RX->Outputs



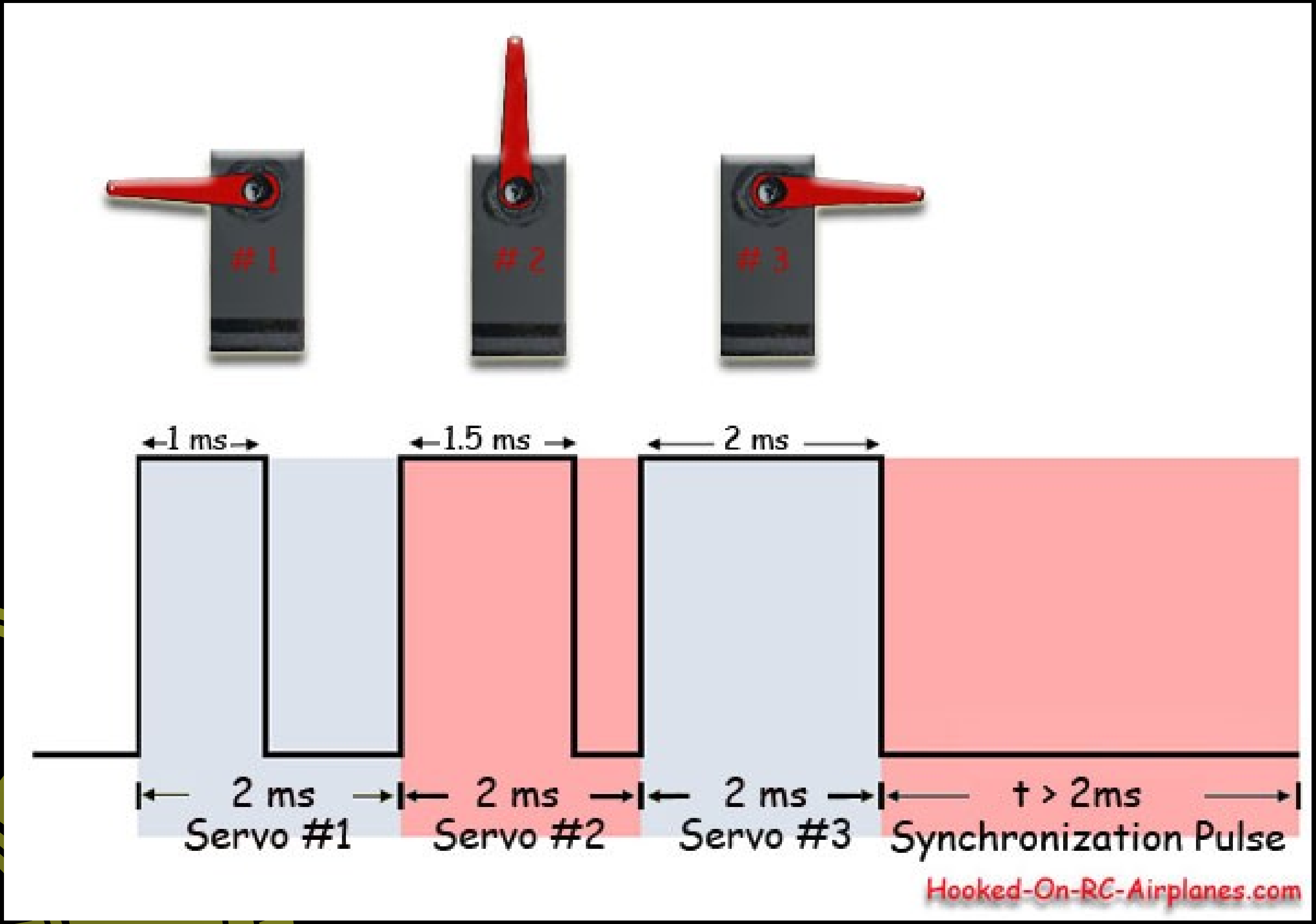
Channels

- Pulse-period: 20ms
- Pulse-length: 1-2ms
- 'center' = 1.5ms

Note

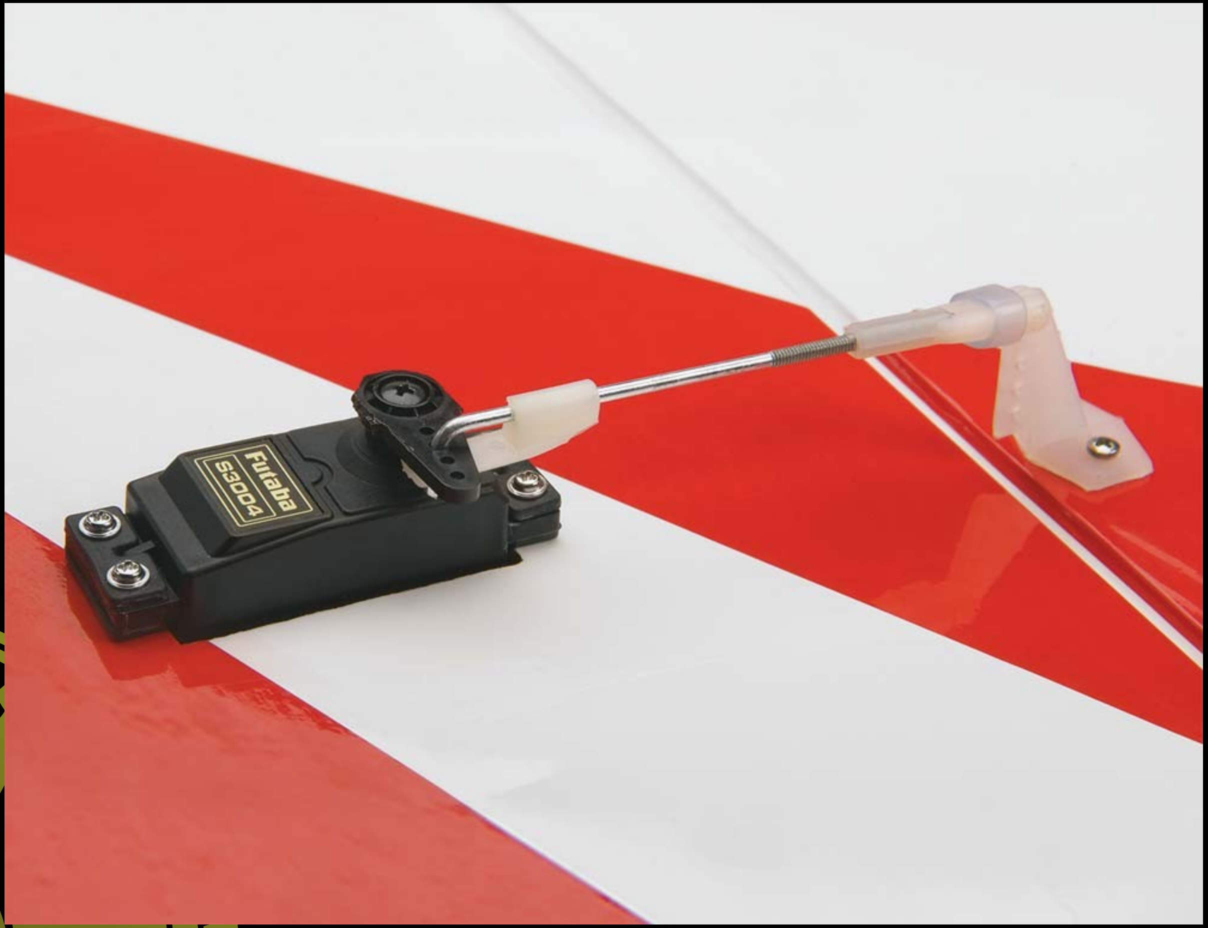
PPM/PWM/etc cargo-culting

Example with servo's



On the topic of Servo's

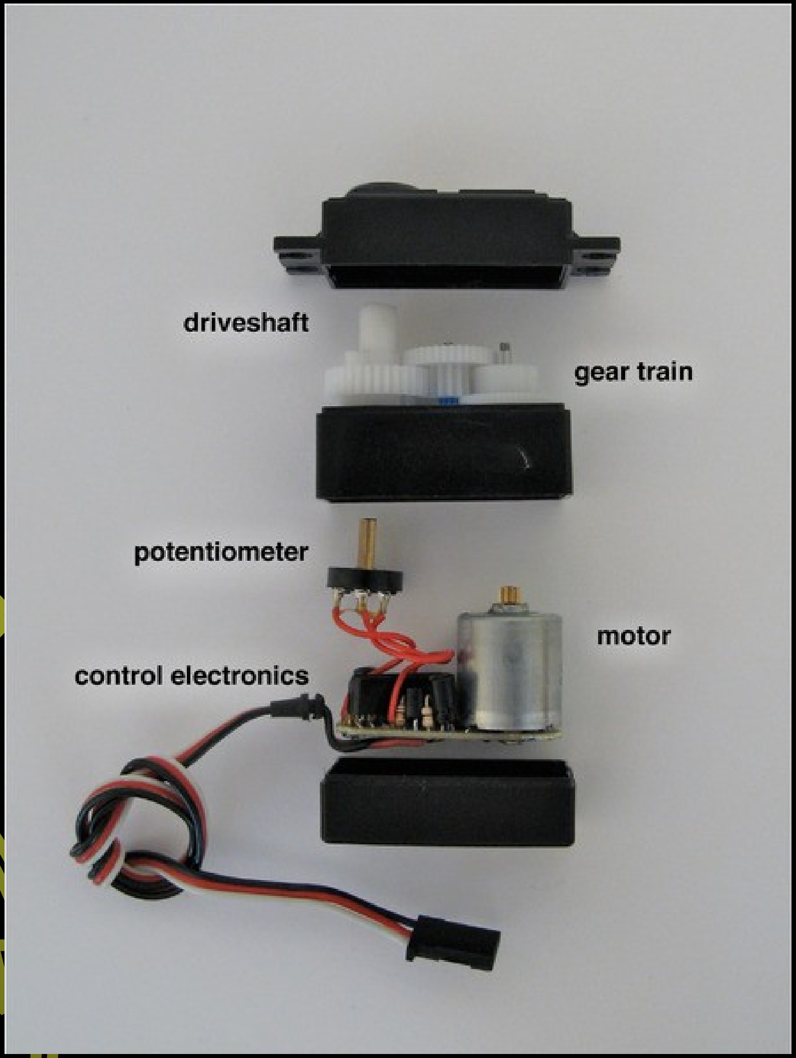
Uses



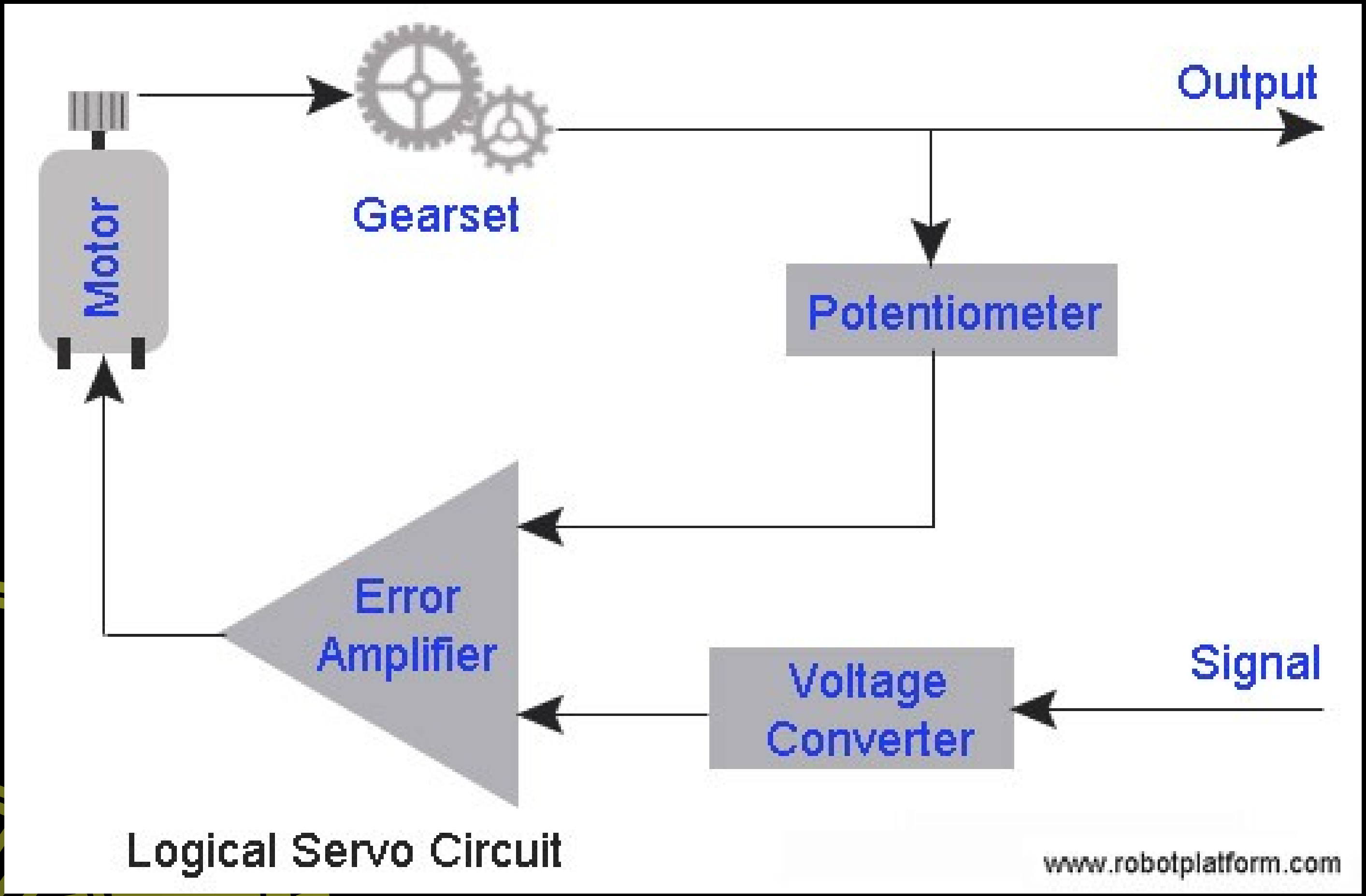
Servo

- Module-based
- Actuator (rotate/slide/escapement)
- Cable with +, - and Signal

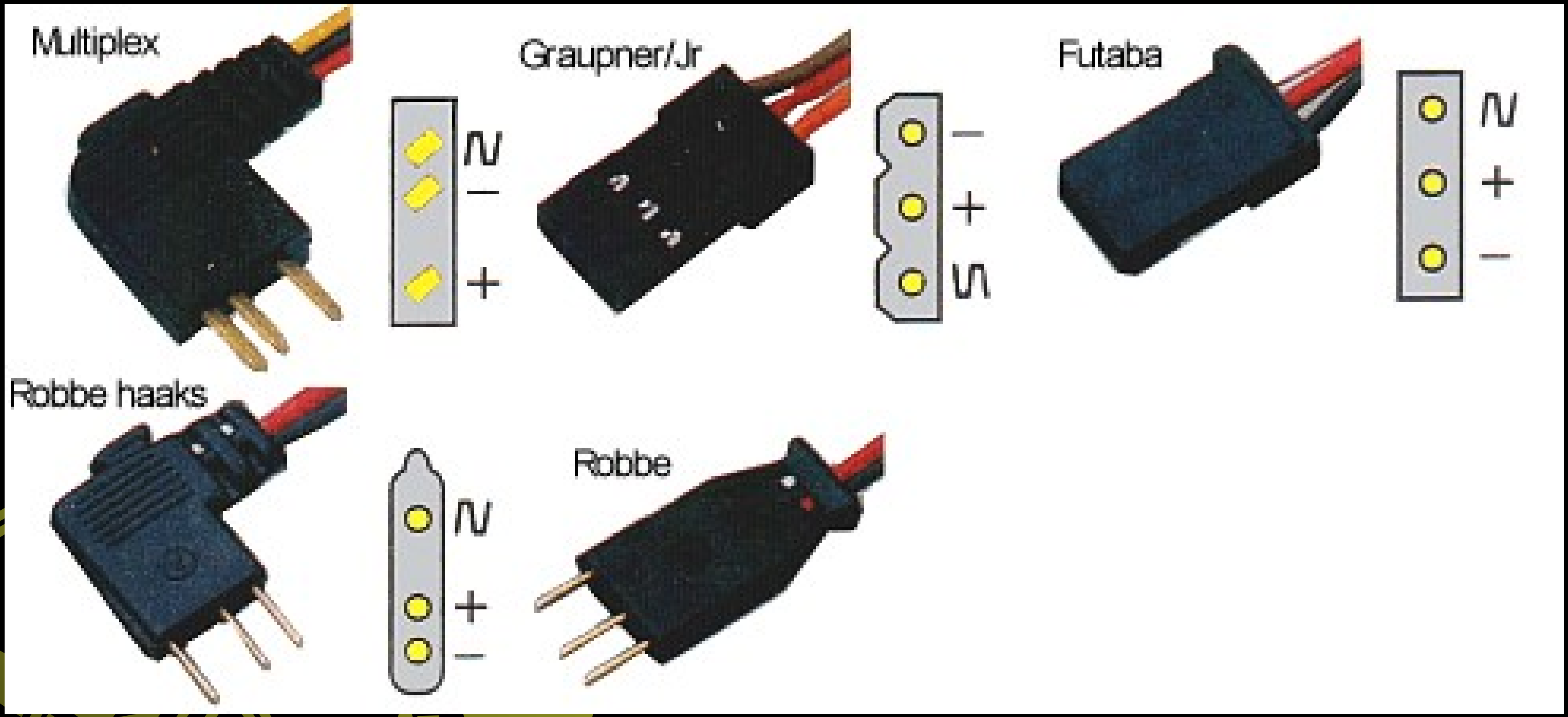
Explode!


driveshaft
gear train
potentiometer
motor
control electronics

Block diagram



Plugopalypse



More uses

- Wiggle flaps
- Flap rudders
- Whinch whinches
- Release the hot lava
- Throttle the throttle

On the subject of: ESC

Old days

- Joe Blow moves stick
- TX sends to RX
- RX controls Servo
- Servo controls pot-meter (or gas-motor)
- Potmeter controls motor
-
- Profit

Potmeters

- Problematic
  - Inefficient
  - No feedback/checks
  - No advanced stuff (stall-detect, restart)

Nowadays

- Modern Electronic Speed Controller
  - Shrinkwrapped PCB with Atmega8 or Silabs uC
  - Inputs: Battery-power + RX-control
  - Outputs: Motor-power, RX-power (!)
  - Uses FETS for control
  - Contains (U/S)BEC !

BEC INTERMEZZO

- **B**attery **E**liminator **C**ircuit
  - Battery-power IN
  - Clean 5v power OUT
  - Connects POWER to RX!
  - Linear BEC = LM7805 (stable, inefficient)
    - See http://wiki.techinc.nl/index.php/78xx_power_supply
  - SBEC = Switched Mode PSU (use only 1!)
  - UBEC = Universal BEC (nonsense term)
  - No BEC ? → OPTO(coupler) (isolated)

Classic motor control

- Put power on motor
- Power magnetizes coil
- Coil pushes against magnets (or other coils)
- Axle turns
- 'brush' contact 'switches' to new coils
- Simple... , but...

Classic motors

- Brush-based design
  - Degrades over time
  - High speeds problematic
  - Arcing/fusing issues

What if...

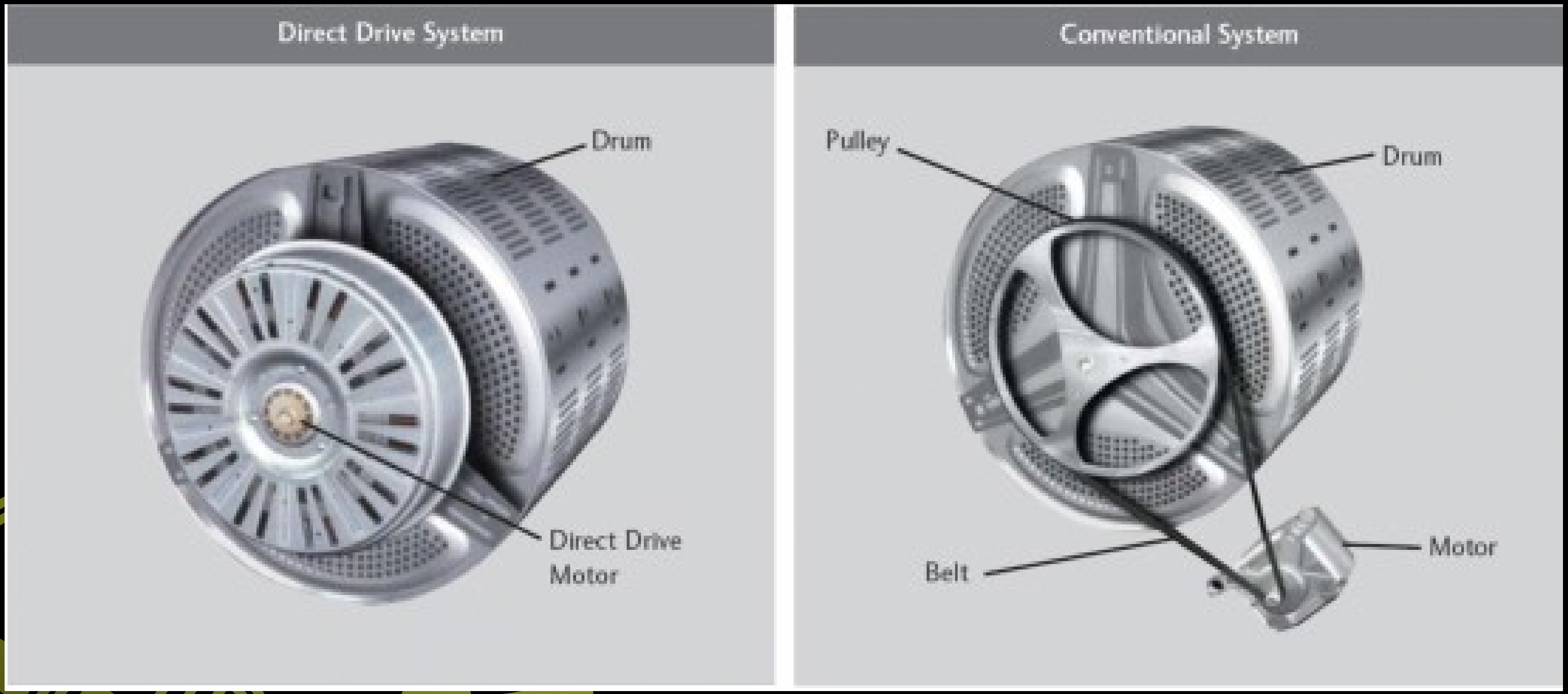
… there was a better way

On the subject of BLDC

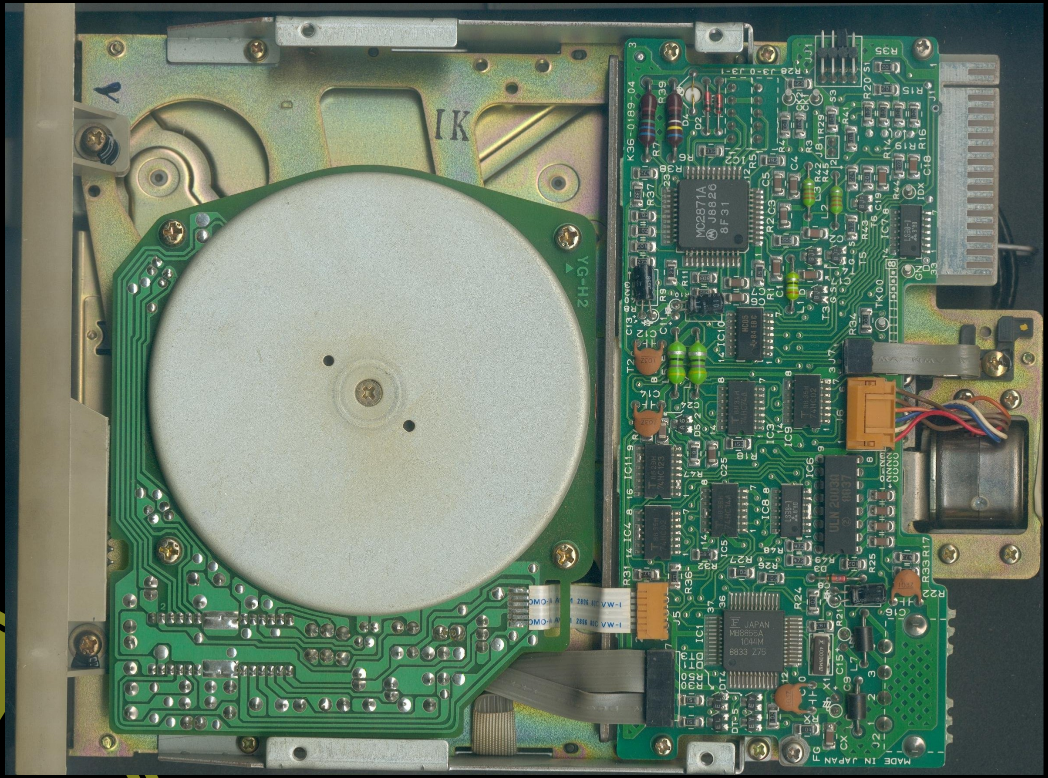
- **B**rush**L**ess **D**irect **C**urrent (Motor)
- No 'powered' moving parts
- No wear on brushes (no brushes! \o/)
- Better/faster control
- Work much like 'stepper-motors'

BLDC: Outrunner

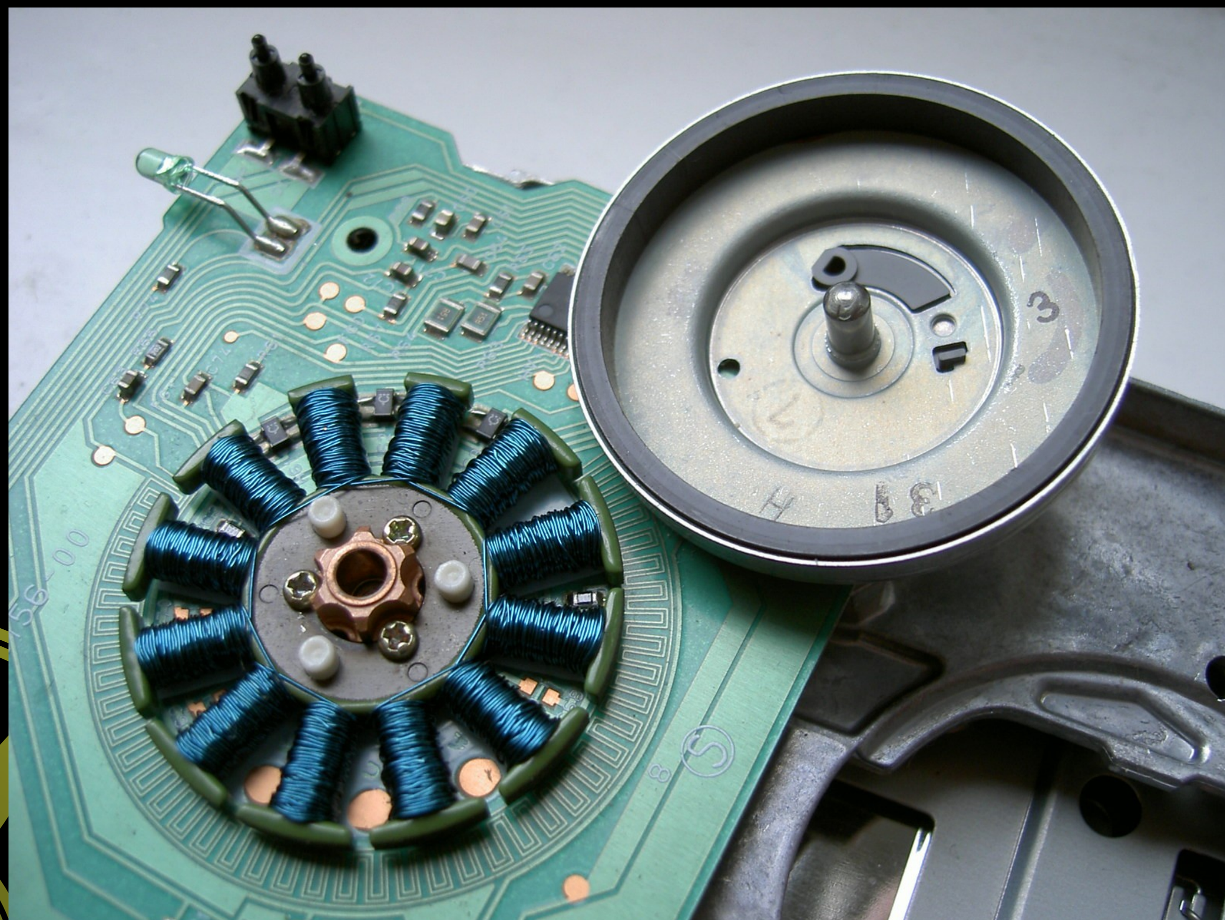
But also...


Direct Drive System
Drum
Direct Drive
Motor
Conventional System
Pulley
Drum
Motor
Belt

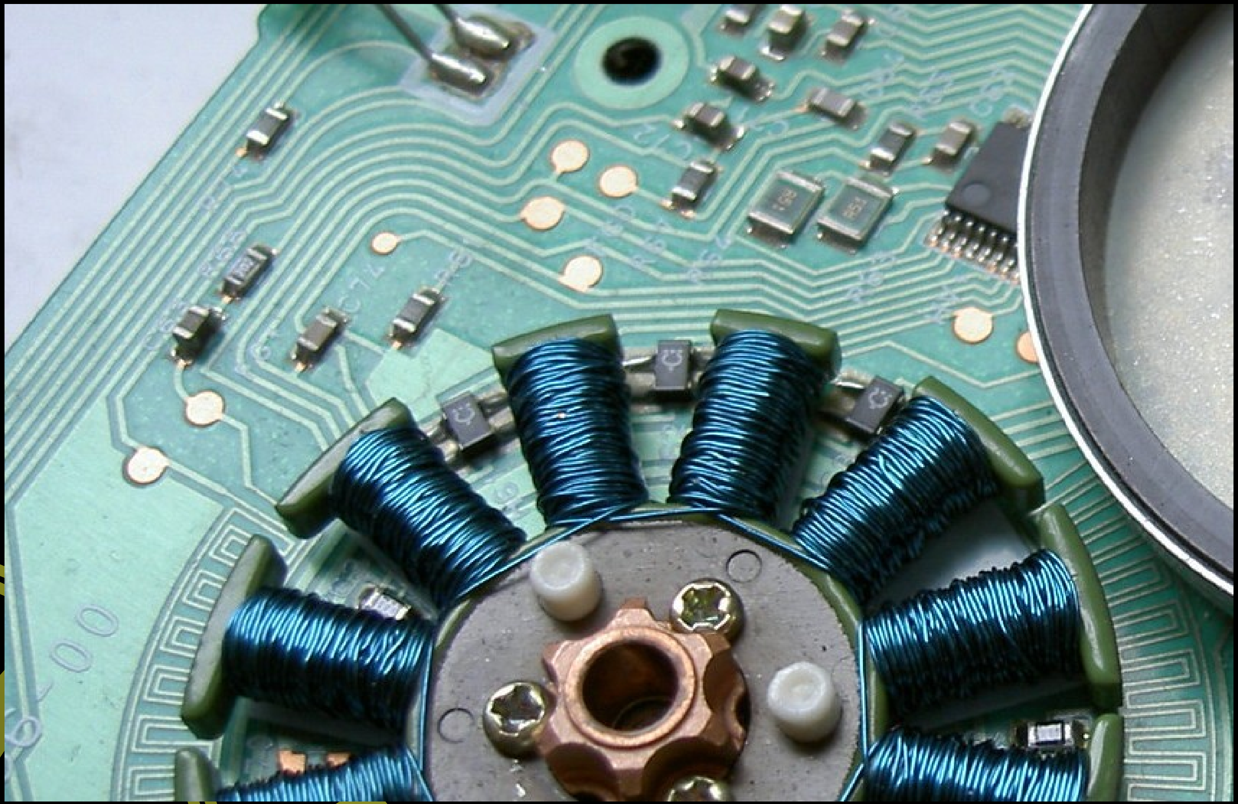
...in fact...



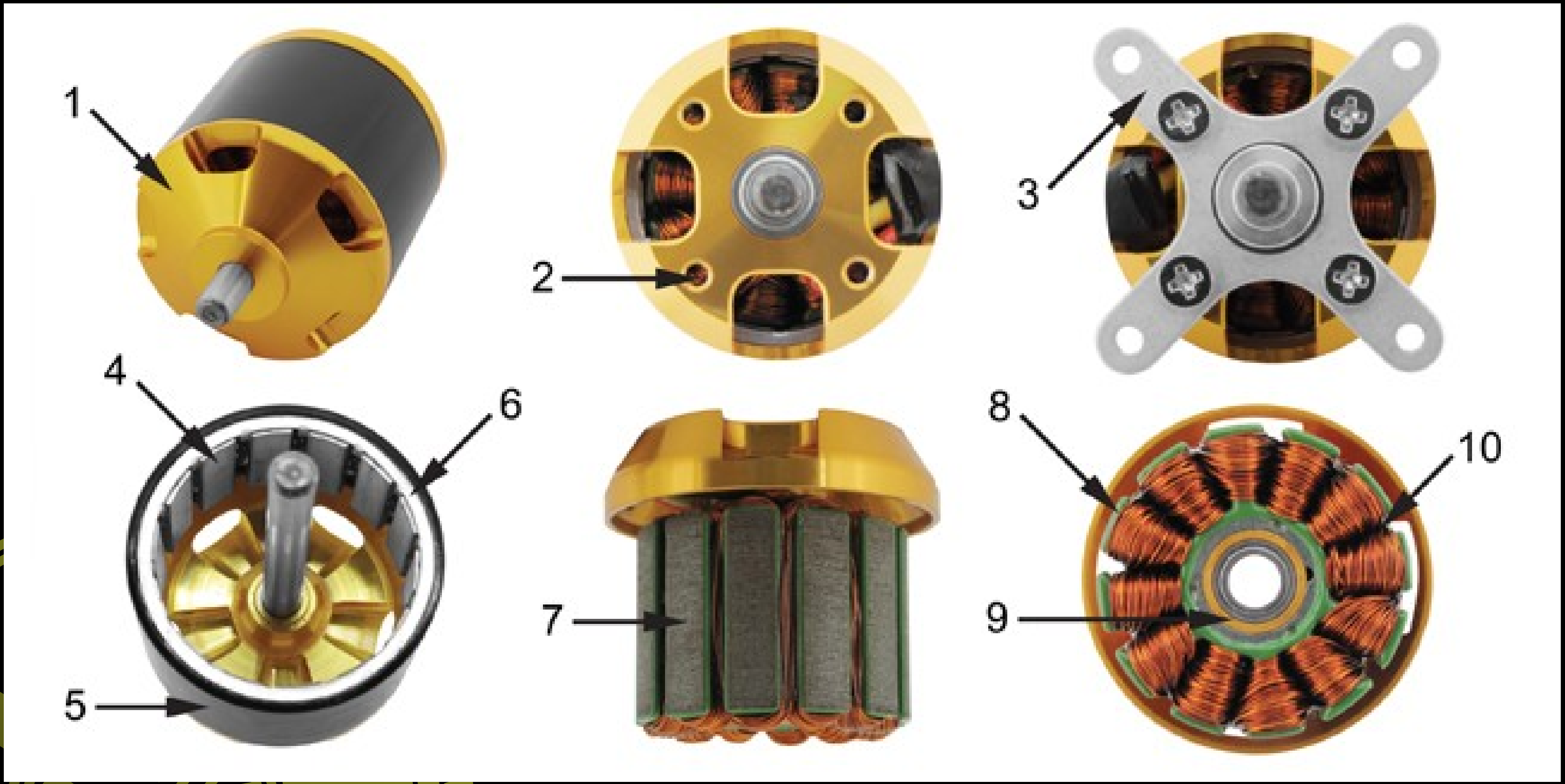
Explained



Enhance!



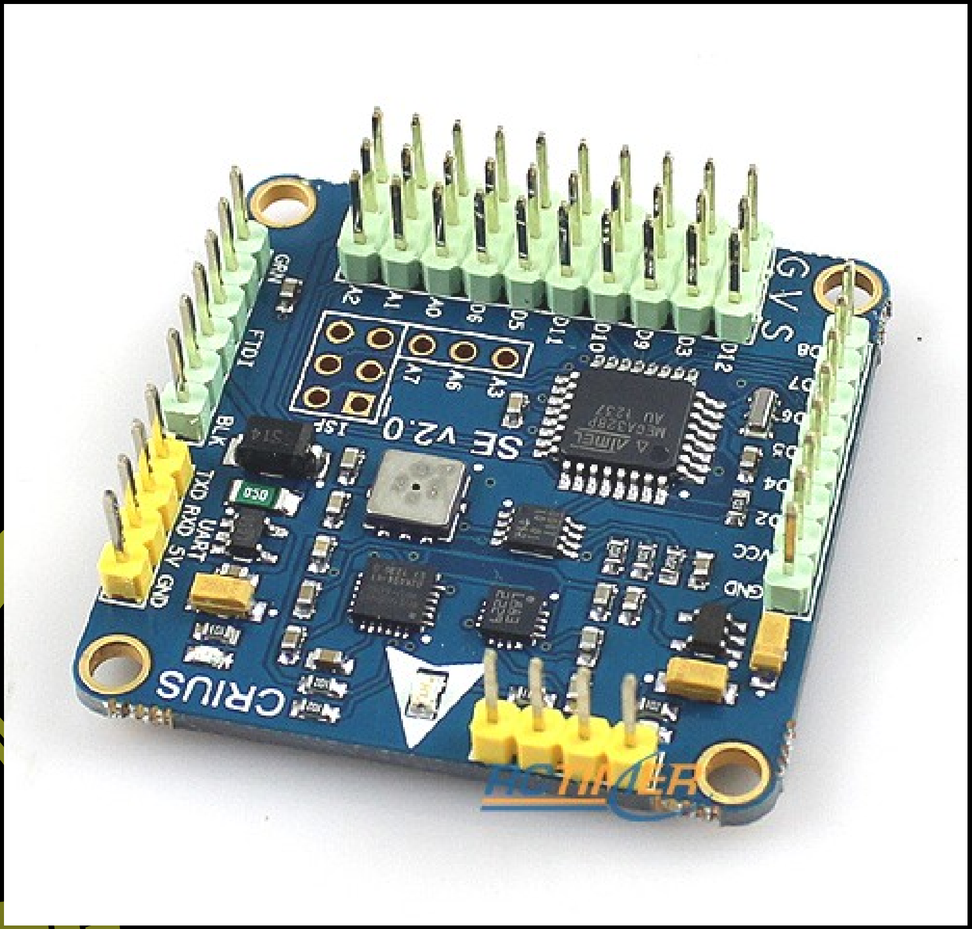
Outrunner



Operation

- Feedback
  - Hall/Back-EMF
- Speed control
  - Control 'coil-sequence speed'
  - Voltage (kV = RPM/Volt)
- Direction control
  - 1-2-3-1-2-3.. or 3-2-1-3-2-1-...
- Stall-detect, speed-detect, Telemetry for FC

On the subject of the FC



On the subject of the FC

- Flight Controller
- Input: RX-signals
- Input: Sensor-signals
- Calculates: error between 'RX' and 'sensors'
- Output: Signals for ESCs or Servo's
- Repeat, repeat, repeat

Flight Controller

- Allows tuning/configuring
- Provides telemetry (for OSD, optional)
- Can drive gimbal, control camera
- Read GPS as sensor-data
- LEDS
- LAZORS
- HOT MOLTEN LAVA

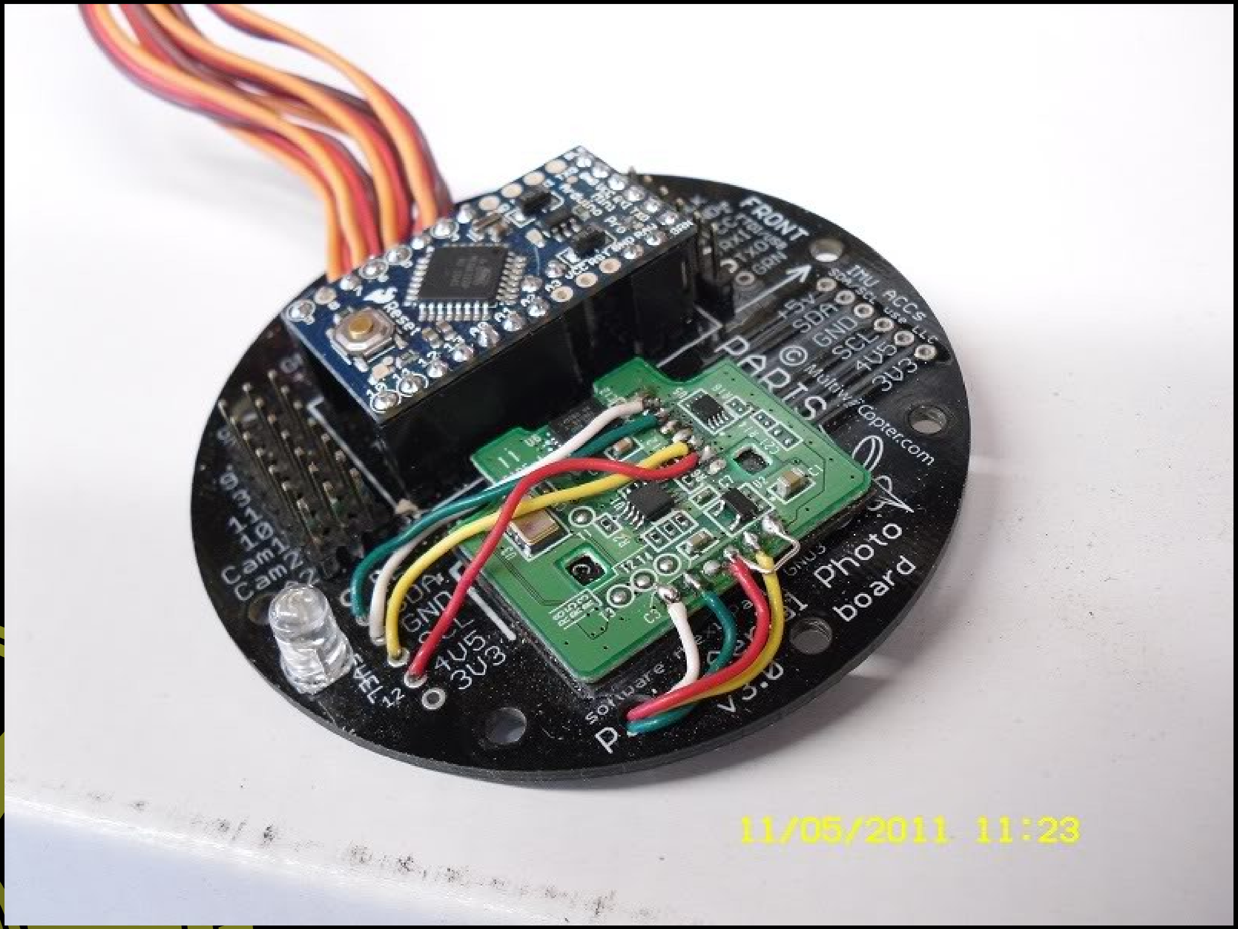
Flight Controller

- Often Atmega(8,168,368, 2560) uC
- Gyroscope (mems)
- Accelerometer (mems)
- Servo-inputs from RX (but also power TO RX)
- Servo-outputs for ESC/Servo/Lava

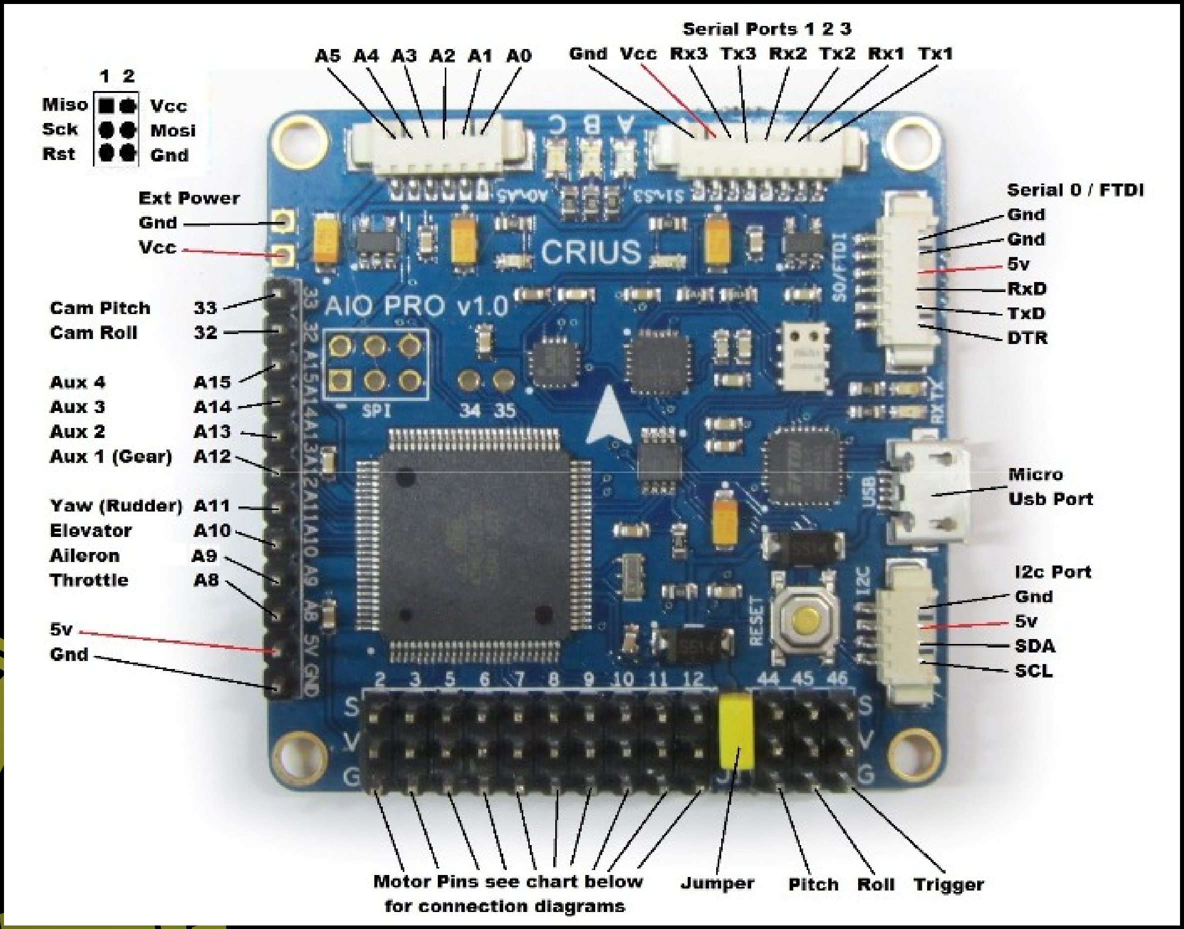
Flight controller

- But also:
  - Outputs for Serial (GPS, BT, Mavlink (osd))
  - Output for extra buses (i2c, SPI, etc)
  - Buttons/potmeters/screens
  - Ping-sensor
  - Optical flow sensor
  - Micro barometer

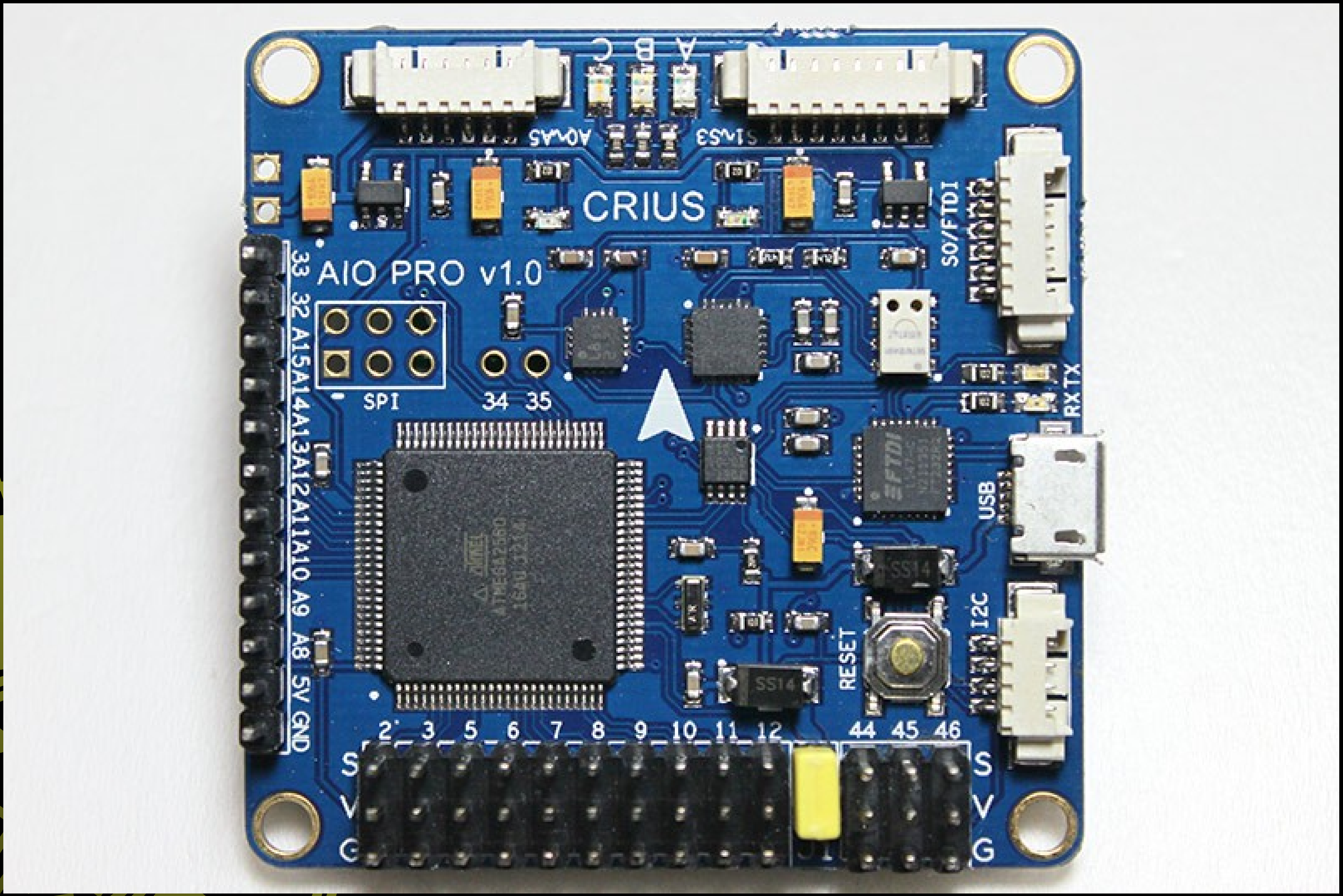
Examples: Wii Motion



Crius AIO Pro



Gyro, Accel, Compass, Baro



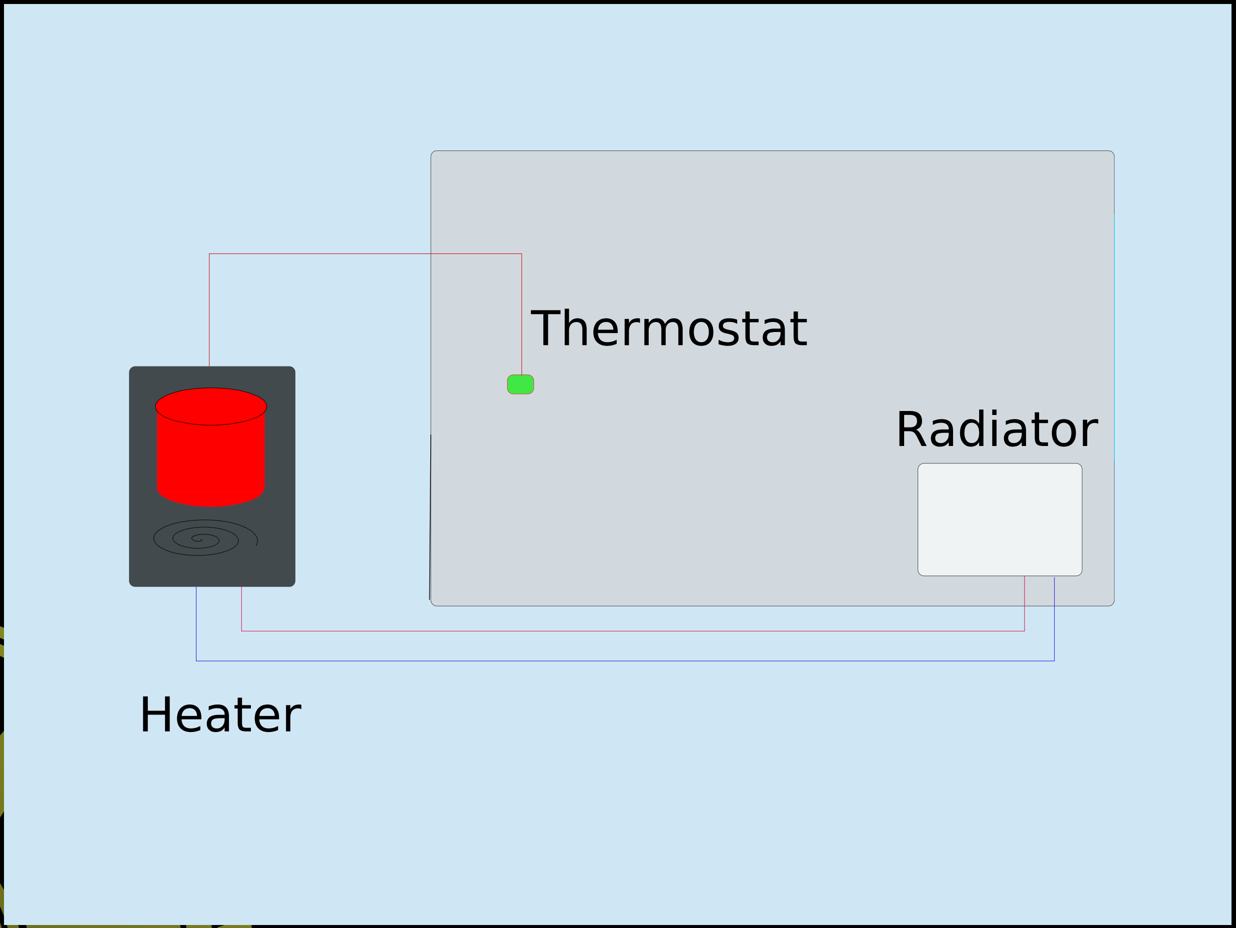
But how...

- Control loops
  - If 'input' = 'fly right' and 'sensor' =! 'fly right' then
  - Adjust outputs
  - Else
  - Do nothing
  - Wash, rinse, repeat

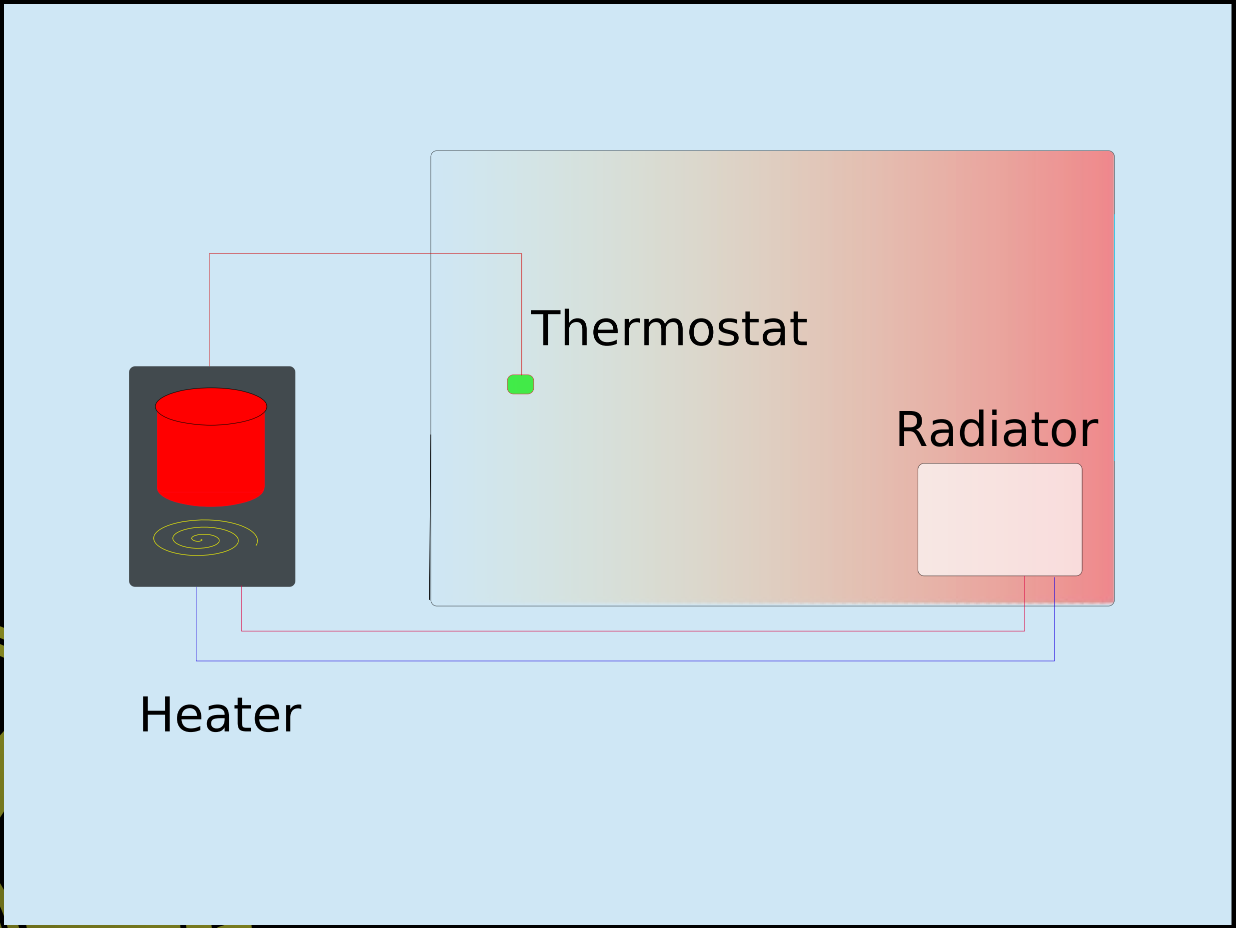
But.. gently

- Proportional adjustment
  - Error between 'input' and 'sensors' small ?
  - Adjust lightly
  - Error between 'input' and 'sensors' big ?
  - Adjust sharply

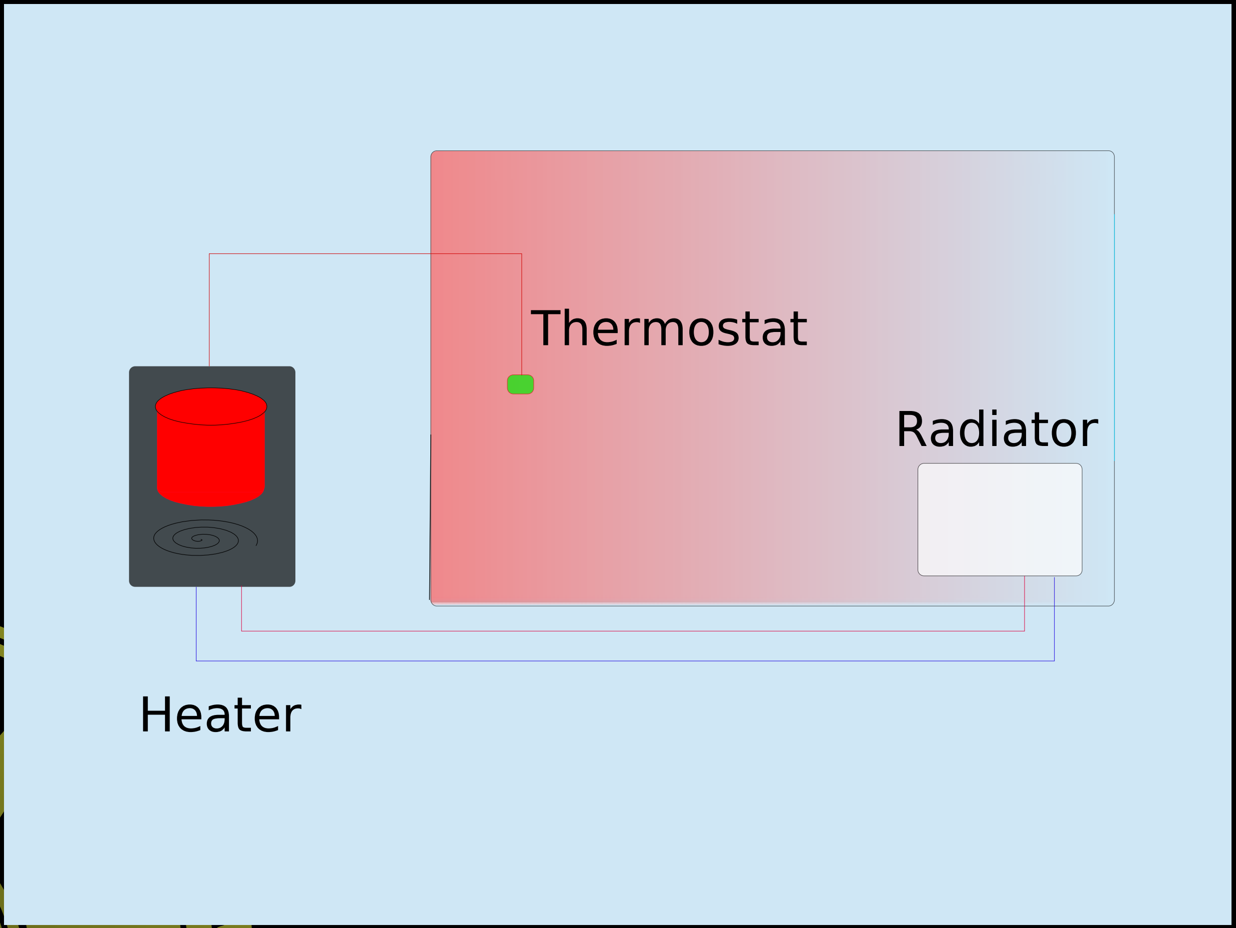
Hot Topic



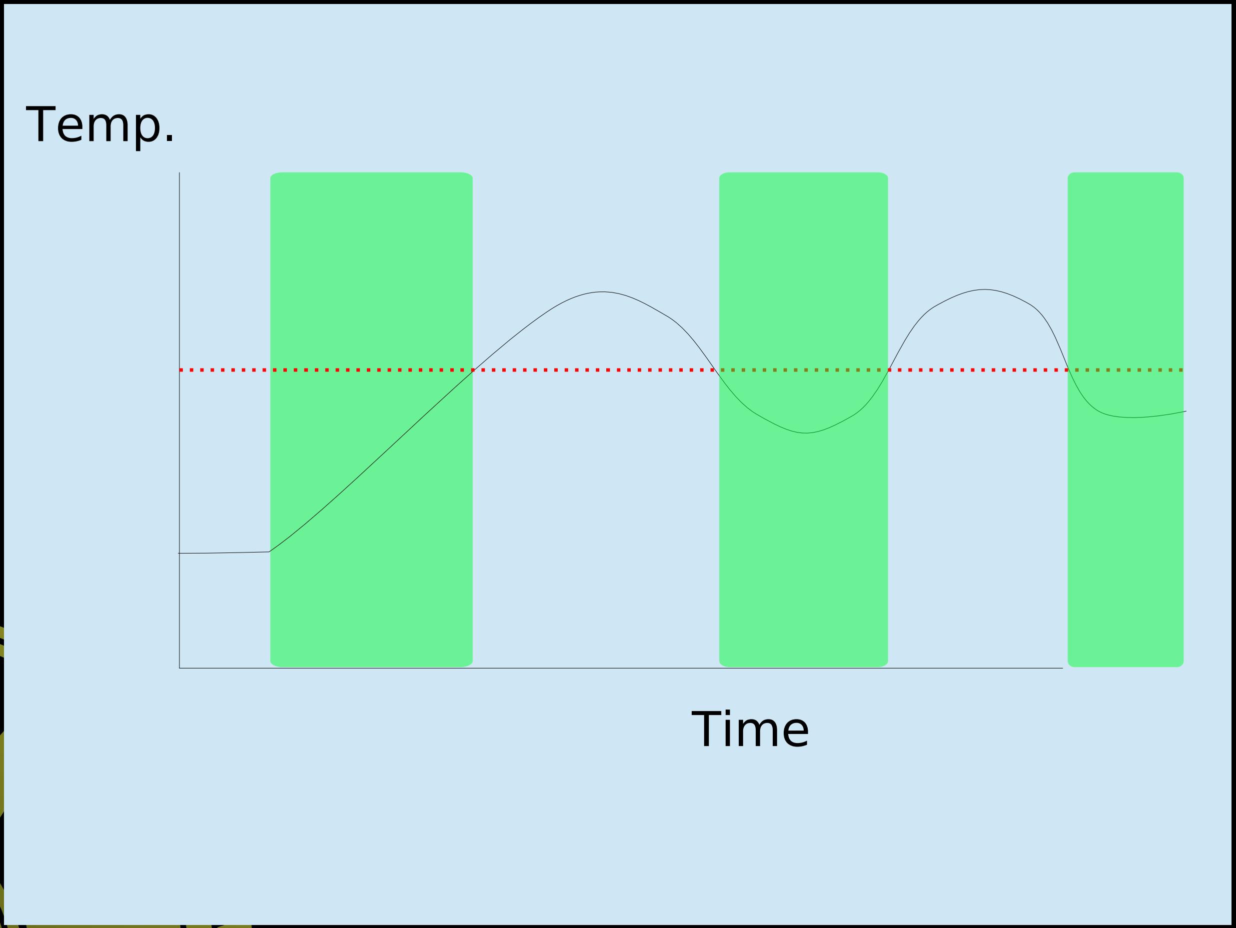
The heat is on!



But wait...



Too Hot, Too Cold



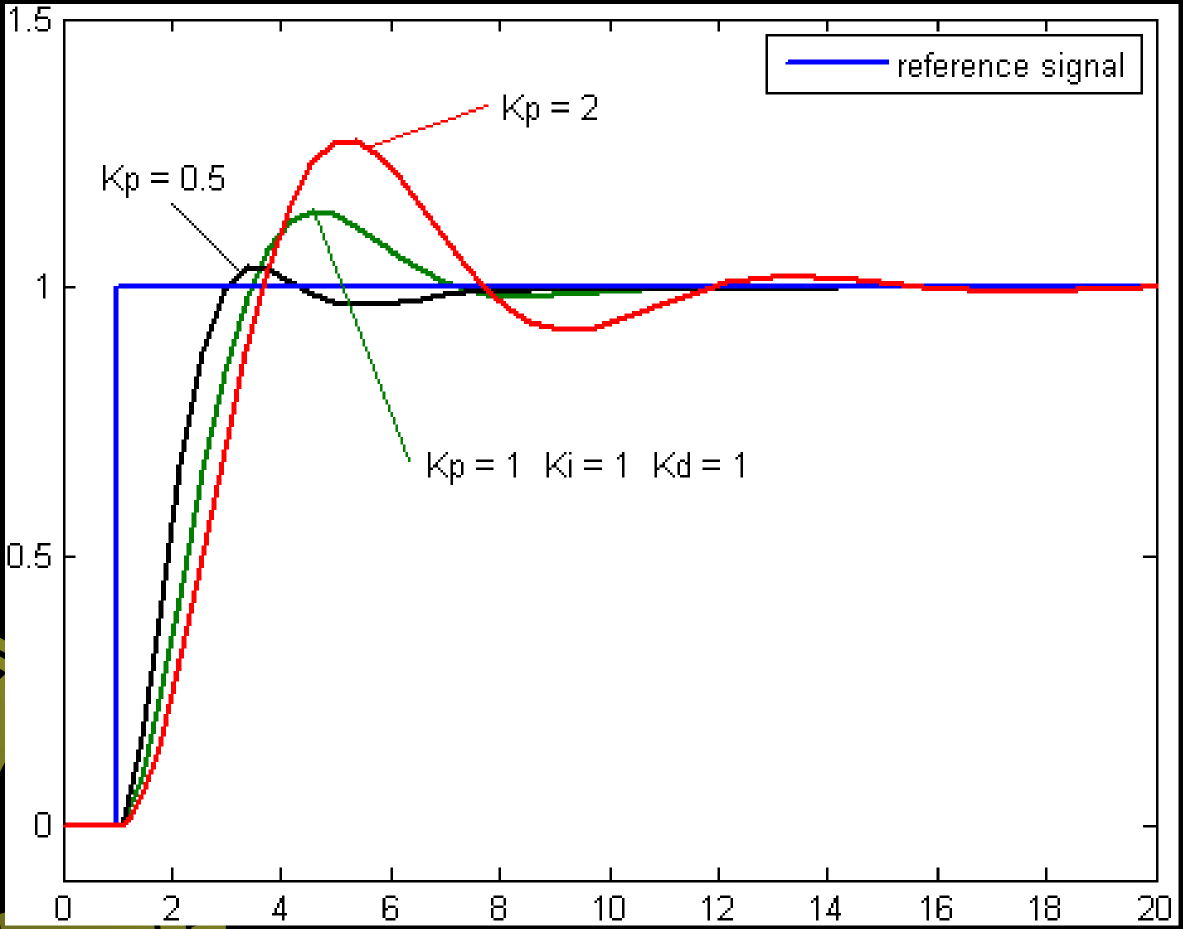
PID Intermezzo!

- <u>P</u>roportional
- <u>I</u>ntegral
- <u>D</u>erivative
- $$\mathrm{u}(t) = \mathrm{MV}(t) = K_p e(t) + K_i \int_0^t e(\tau)\, d\tau + K_d \frac{d}{dt} e(t)$$
- PID

Fuck math; let's go flying

- Proportional, depends on 'current error'
  - If possible, don't just switch on and off
  - If not, use pulse-width-modulation
- Integral, depends on 'sum of past errors'
  - Teaches the controller how 'fast' your control-loop reacts
- Derivative, prediction of future errors
  - Tries to be smart about what is sees and what it knows. Not used often in multi-rotors

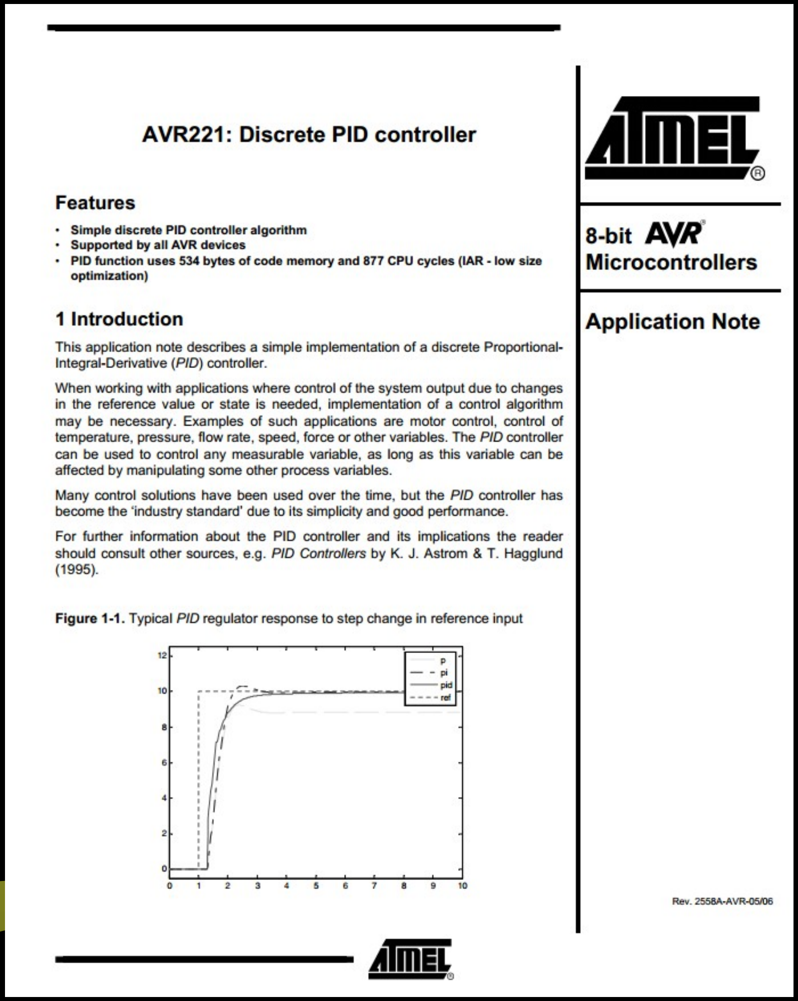
PICS || GTFO



reference signal
Kp = 2
Kp = 0.5
Kp = 1 Ki = 1 Kd = 1
1.5
1
0.5
0
0
2
4
6
8
10
12
14
16
18
20

For frying

For flying



AVR221: Discrete PID controller

8-bit AVR Microcontrollers

Application Note

Features

- Simple discrete PID controller algorithm
- Supported by all AVR devices
- PID function uses 534 bytes of code memory and 877 CPU cycles (IAR - low size optimization)

1 Introduction

This application note describes a simple implementation of a discrete Proportional-Integral-Derivative (*PID*) controller.

When working with applications where control of the system output due to changes in the reference value or state is needed, implementation of a control algorithm may be necessary. Examples of such applications are motor control, control of temperature, pressure, flow rate, speed, force or other variables. The *PID* controller can be used to control any measurable variable, as long as this variable can be affected by manipulating some other process variables.

Many control solutions have been used over the time, but the *PID* controller has become the 'industry standard' due to its simplicity and good performance.

For further information about the PID controller and its implications the reader should consult other sources, e.g. *PID Controllers* by K. J. Astrom & T. Hagglund (1995).

Figure 1-1. Typical *PID* regulator response to step change in reference input

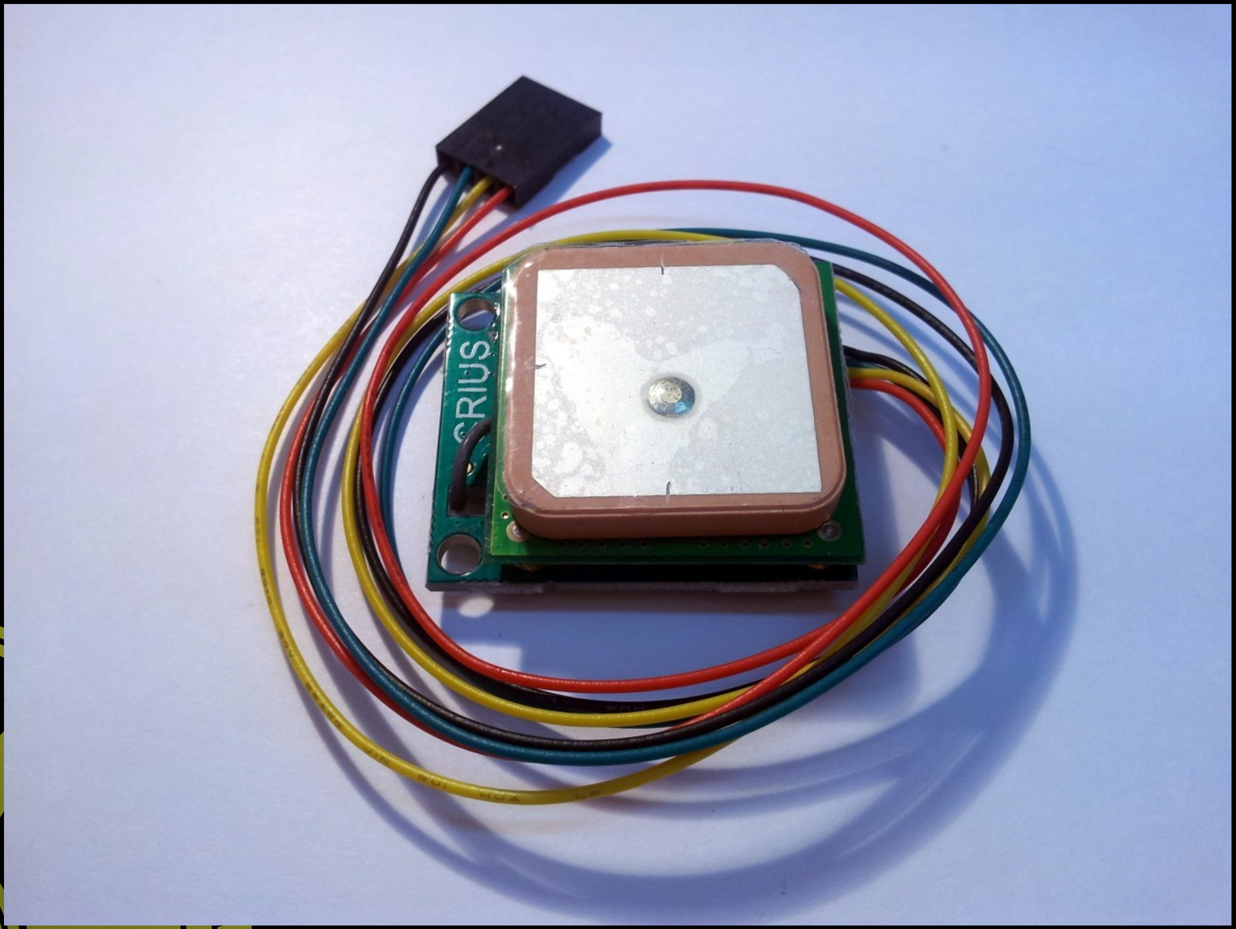
Rev. 2558A-AVR-05/06

Phew that was boring

But important

You will thank me later

GPS , YAY

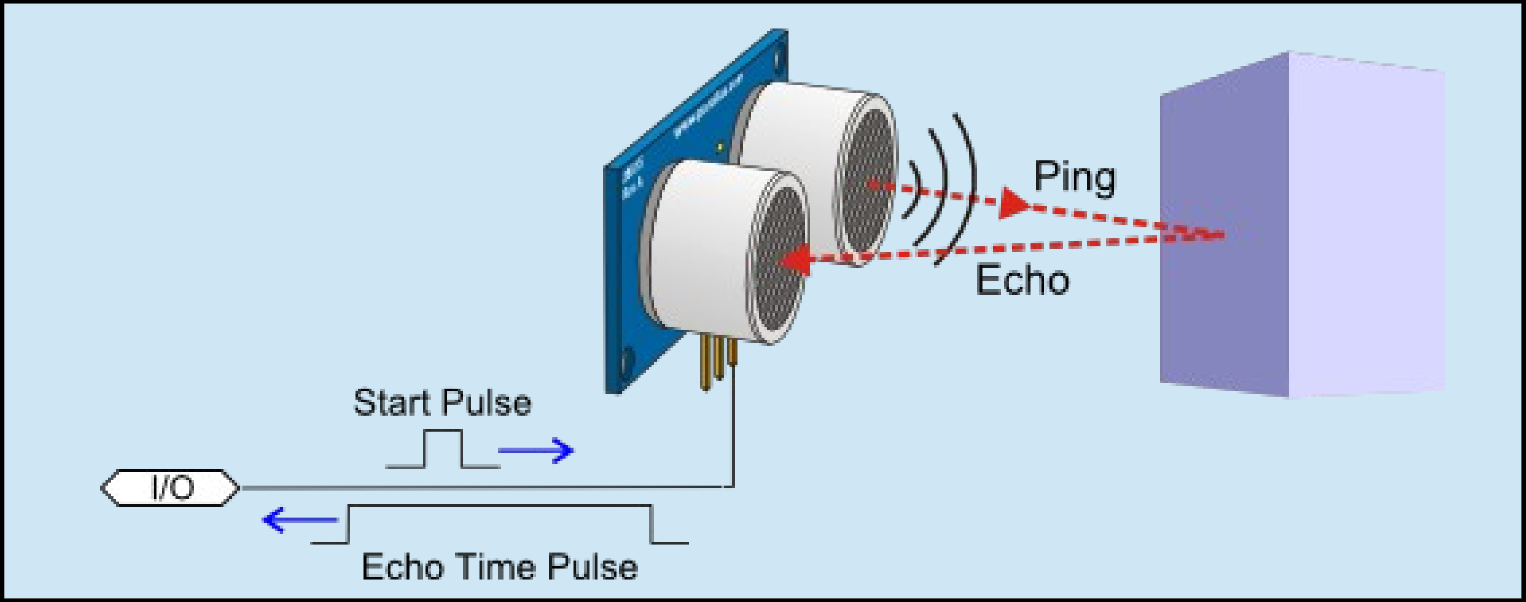


GPS, uses

- Pro: Absolute position
- Con: too slow and innacurate
- Con: no tilt

- Use: Path planning
- Interface: Serial, I2C
- State of the art: 10hz Update, 115k2 BPS

Ping-sensor

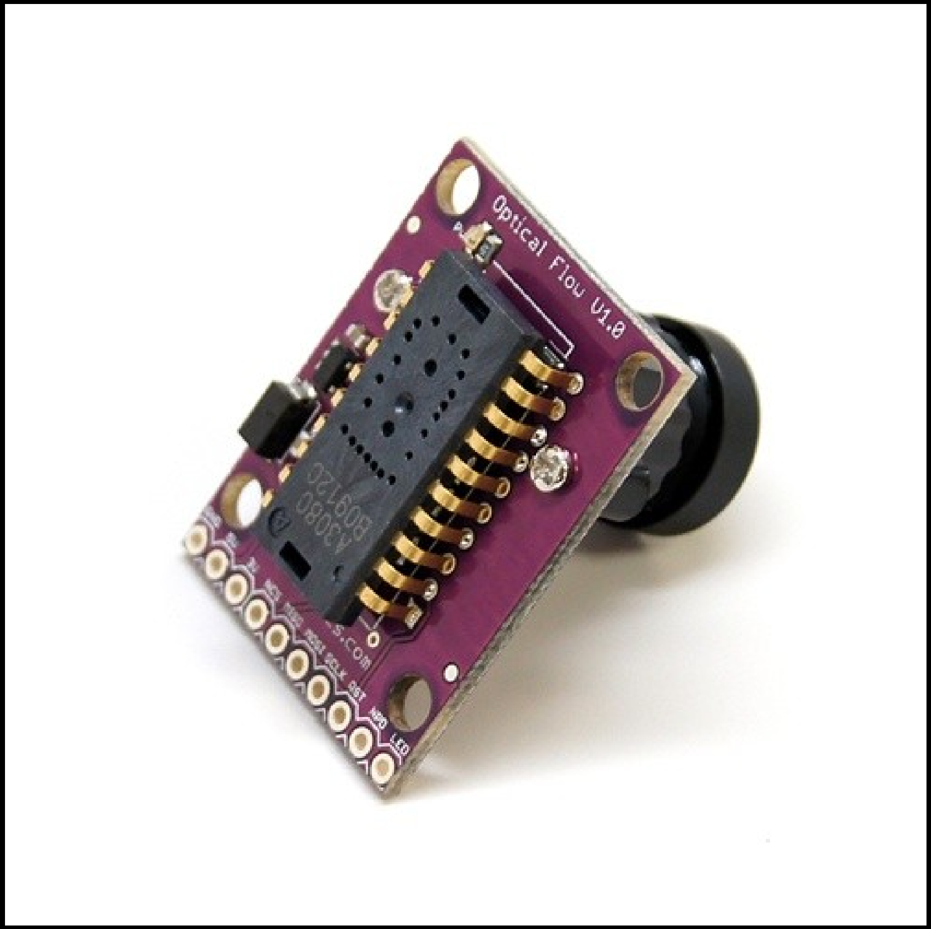

Ping
Echo
Start Pulse
I/O
Echo Time Pulse


28015
REV B
ACT
PING)))
TM
GND
5V
SIG

Ping Sensor

- Pro: Great for collision avoidance
- Con: Useless for anything else
- Connection: analog pins
- Used in: Parrot AR

Optical Flow sensor



MoCap

Check Mah Bling

Cameras in the space

GREAT

Back to DIY

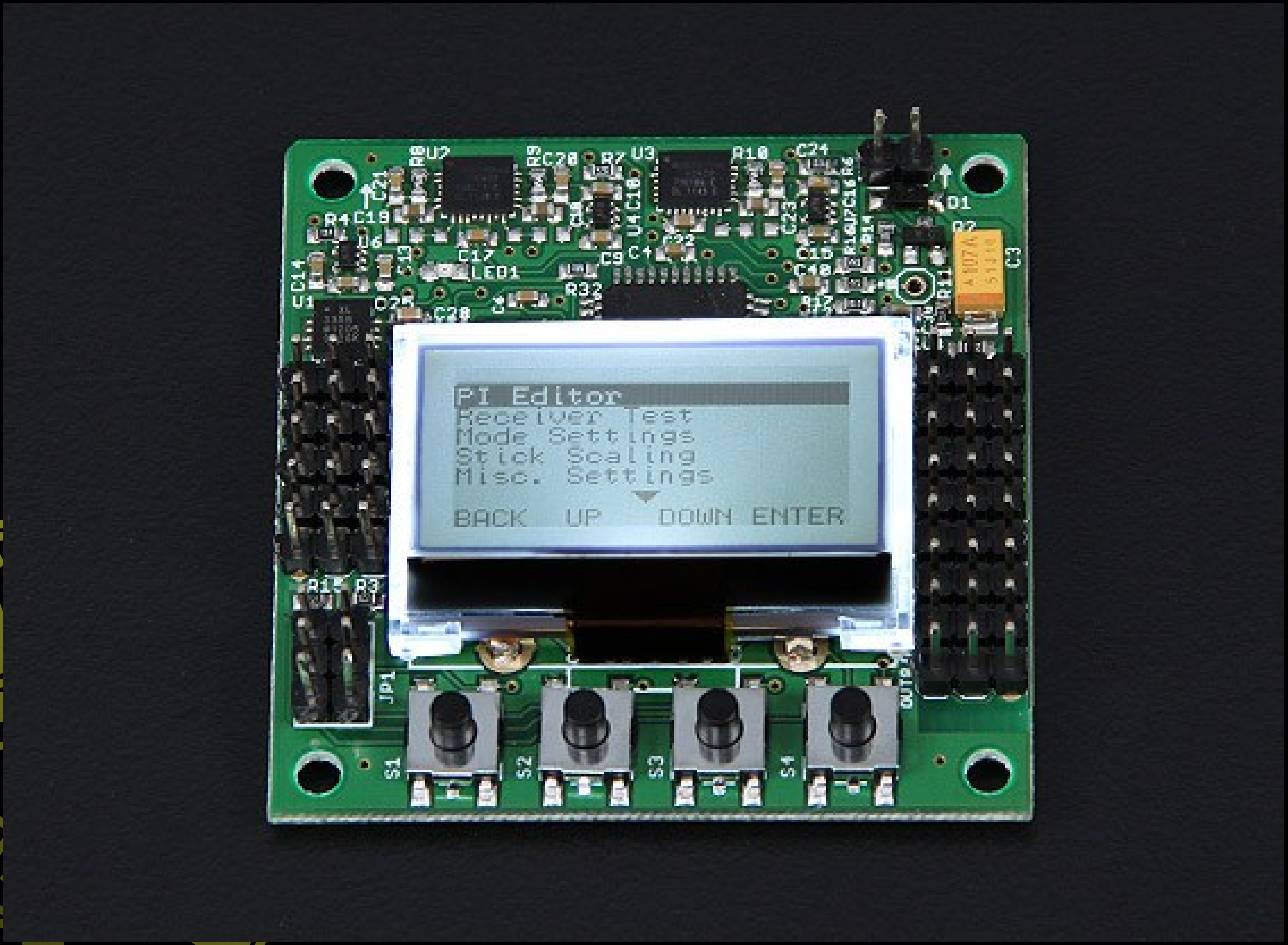
Ardu/Atmega-boards

- Arduino-based/inspired.
  - MultiWii (MegaPirate), FlyDuino, ArduPilot
  - Atmega based
  - Programming: Serial/USB via Arduino software
  - Pro: Well understood, cheap
  - Pro: Great selection available
  - Con: limited speed

KK-boards

- KaptainKuk
  - V1.0: simple, limited use
  - V2.0: Limited feature-set, easy to configure
    - Screen
    - Buttons
    - Beeper
    - Voltage-level measurement
    - 8 channels
    - Camera-rig levelling/Servos
    - Con: No extensions (GPS, BT, etc), HobbKing only

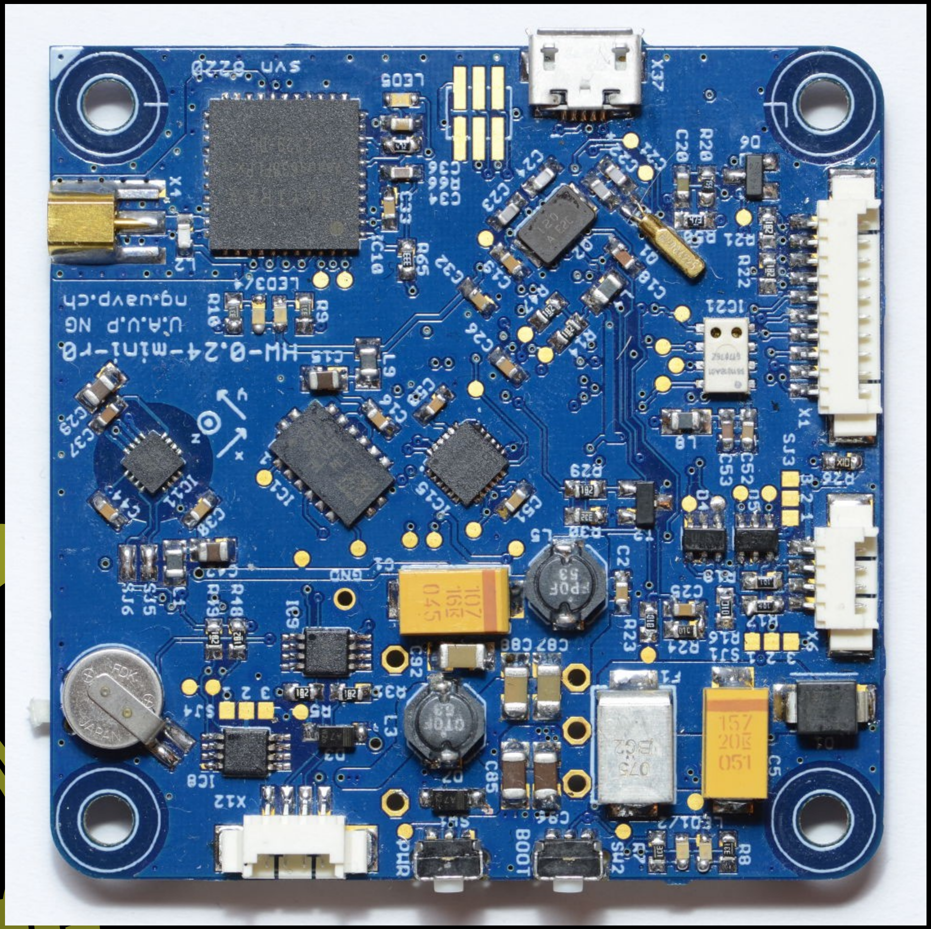
KK2.0, example



NG.UAVP.ch

- Next Generation Universal Video Platform
  - LPC2148 ARM7 core @60Mhz
  - (Atmega 644 just for extra functions)
  - Pro: Open source
  - Pro: Super extensible
  - Con: harder to work with
  - Con: i2c-based ESC's required
  - Con: expensive (199 euro and up)

ng.uavp.ch 0.24-mini



AeroQuad

- Open source
  - STM32 @168Mhz or Atmega2560
  - Pro: flexible, capable
  - Con: expensive-ish ($149 for STM32)

Cellphones =! cameras

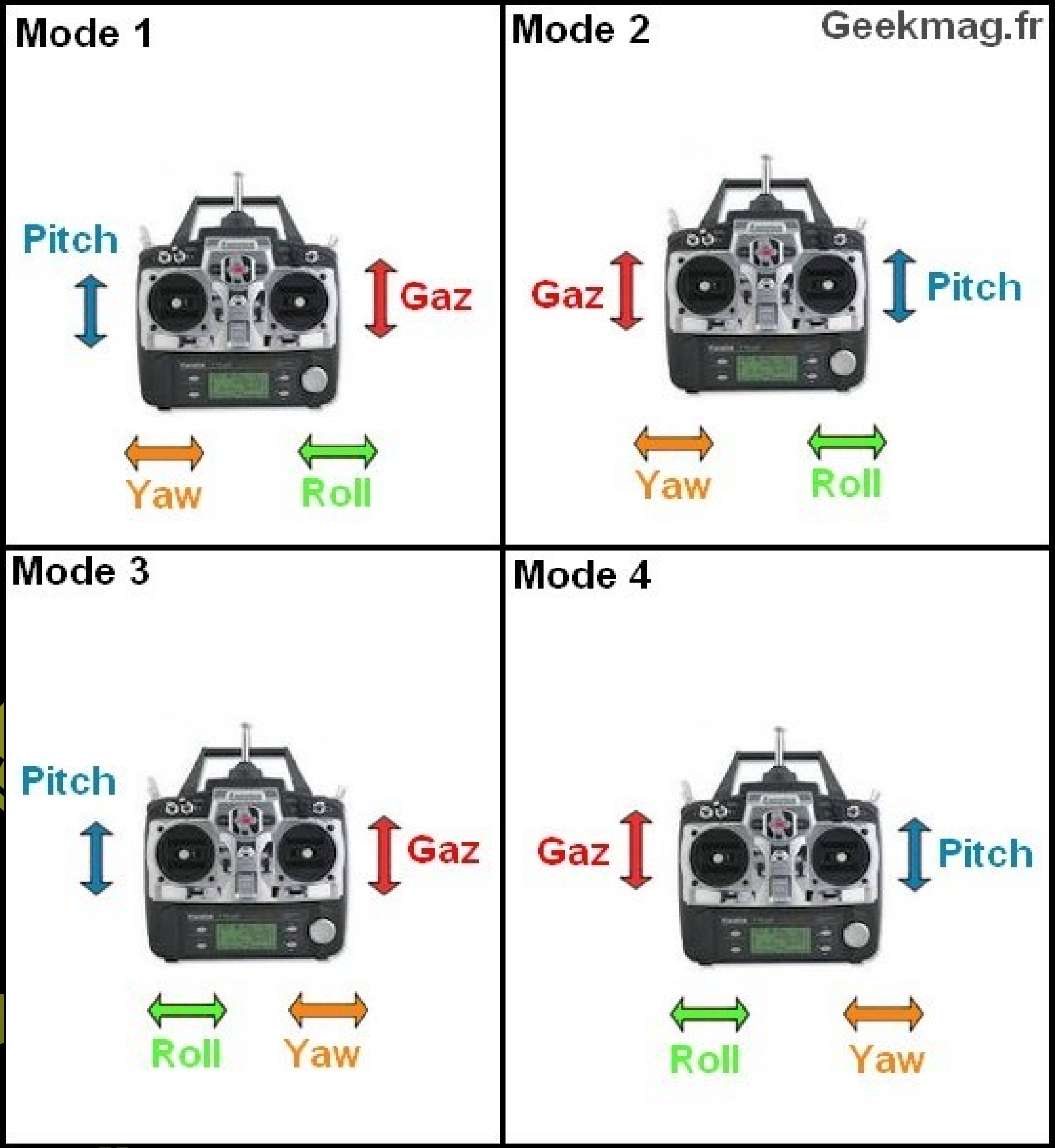
And more...

… and each year...

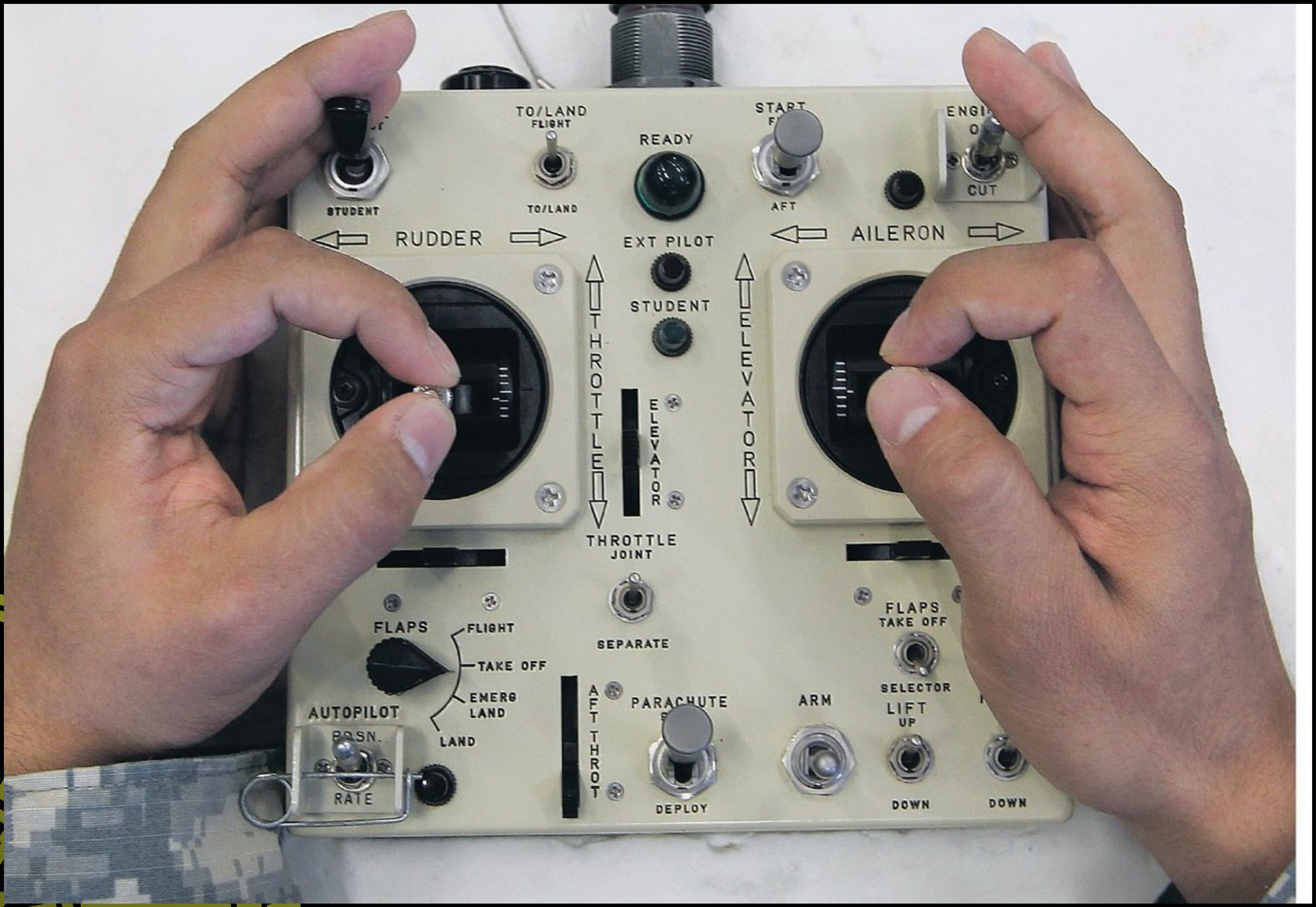
What you also need to know

- Transmitter
  - Mode 1
    - Left: Elevator , Rudder
    - Right: Aileron, Throttle
  - Mode 2
    - Left: Rudder, Throttle
    - Right: Elevator, Aileron
- Short story: Get mode 2 for multirotor

Pics du jour


Mode 1
Pitch
Gaz
Yaw
Roll
Mode 2
Geekmag.fr
Gaz
Pitch
Yaw
Roll
Mode 3
Pitch
Gaz
Roll
Yaw
Mode 4
Gaz
Pitch
Roll
Yaw

Sticks, knobs, buttons



ahem

Adjustments

- Pitch (offset)
- Expo (reactiveness curves)
- Inv/Norm (Channel Inversion/Normal)
- Channel Mixing (Acro, 360, 120.. etc)

Features

- Multi Model Memory
- Screen
- Digital Trim
- PC-programming
- Buddy-link
- Programming-header
- Telemetry/FPV

Phew

Breathe in, out

You made it

BUT WAIT

The point....

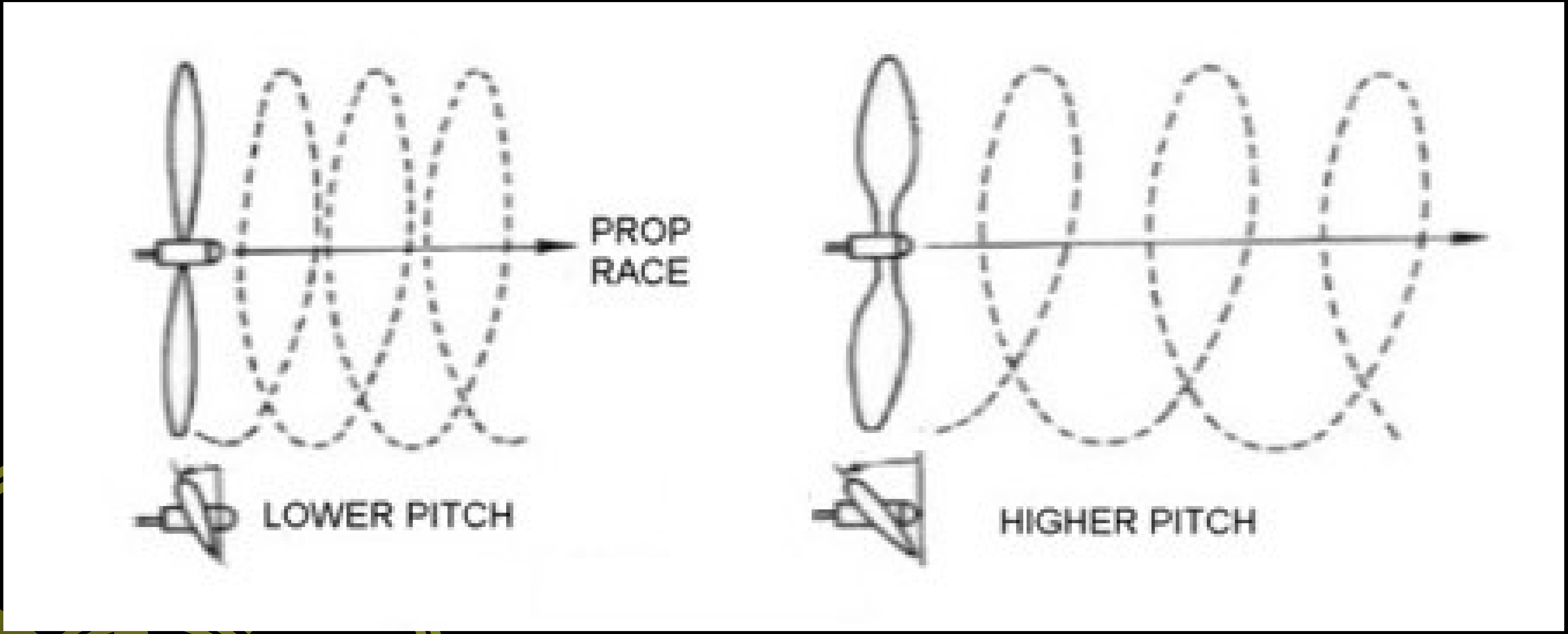
It's about thrust, stupid

- Thrust = lift
- (amount-rotors) * (thrust(motor/propcombo) == (total weight*2)
- English: All motors/props together should produce enough thrust to lift aircraft off the ground at approx 50% throttle

Props to the peeps

- Propellor/Rotor
- Two-bladed, three-bladed, 4-bladed
- Nylon, APC, Carbon
- 'Slowfly' class
- SIZE matters
- PITCH matters

pitch



Note on props

- Buy enough
- Check rotation
- Balance them with tape on upper surface

Flight-time

- Weight leads to
- Lift leads to
- Thrust leads to
- Power leads to
- … weight…

This is Batt-country!

- Lithium Polymer (LiPO)
  - Light
  - High power
  - Low price
  - Explosive
  - Picky
- Do not discharge fully.
- Do not leave charged too long

Batt-crazy

- 3.7v per 'cell'
- 1S = 3.7
- 2S = 7.4
- 3S = 11.1
- 4S = 14.8
- etc

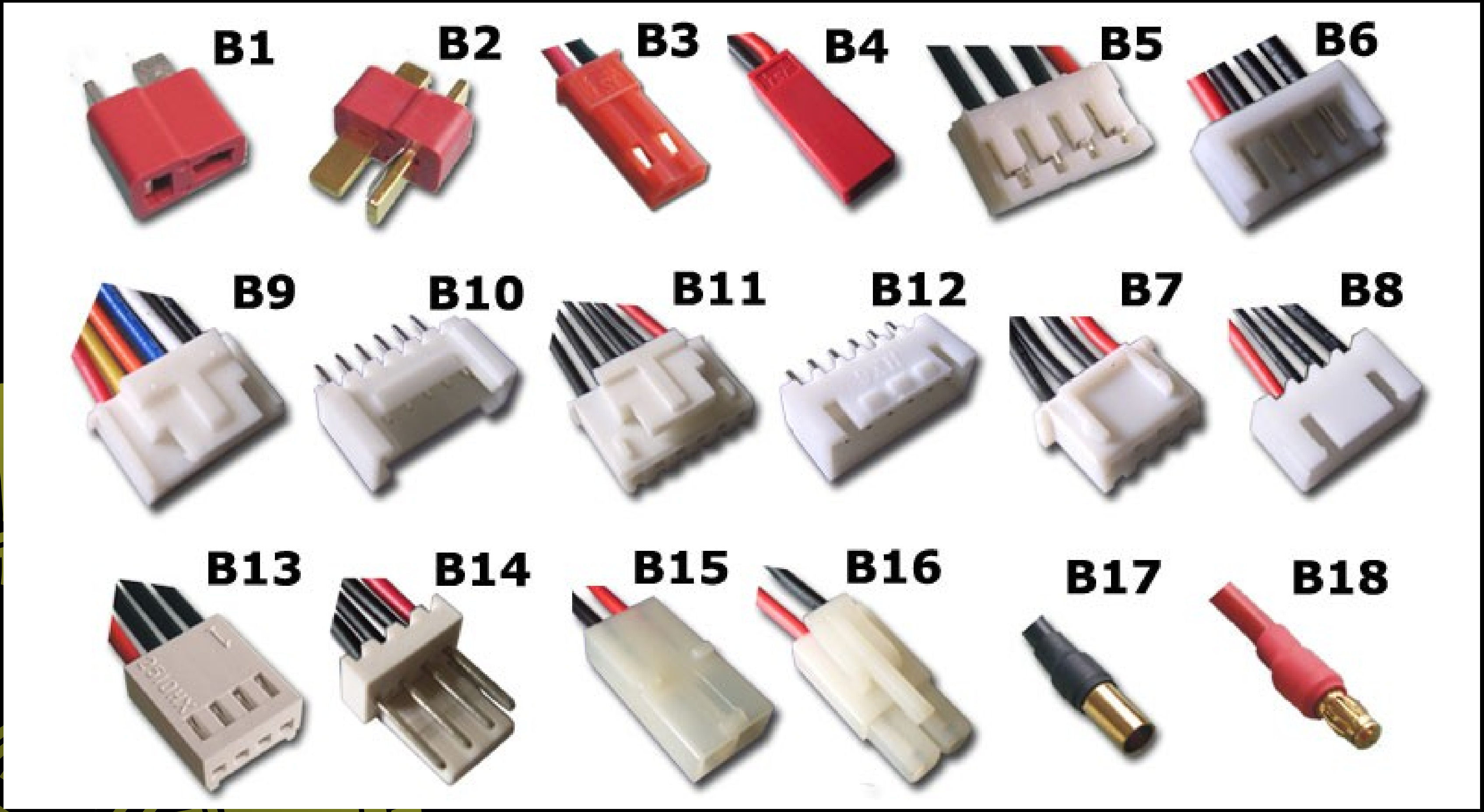
Batt-crazy, 2

- mAh = Milli-Amp/Hour
- 400-40000mAh
- 1C = 1 'charge'
- Discharge C (1C - 40C)
- Charge C (1C – 5C)

Batt-plugs

- Plugs
  - Power plugs:
    - XT60
    - T-Plug/Deans
    - JST
    - Traxxas
  - Balance plugs
    - Pincount = S-number + 1 gnd

Batt-plugs



Charging mah LiPO

Conclusions

- 50% throttle ~= weight
- Max motor-power * motor-count = Max discharge Power (watts)
- (max discharge power)/(battery voltage) = max discharge current (Amperes)
- (max discharge current)/(Battery mAh) < (Battery discharge C)

… and

- Flighttime (hover) ~= (50% max discharge current)/(Battery mAh), in hours
- Flighttime (max lift) ~= (100% max discharge current)/(Battery mAh), in hours
-
- … but

Real life

- Propellor efficiency
- Motors friction
- Motor efficiency
- Heat-loss
- Resistance/Inductance

Studying of kees

Justacopter 1

- Rotors: 4x max 102Watt,1100kV, 8045 prop
- Lift: max 2kg (approx 500g/motor)
- Model weight: 600g
- Battery weight:
  - 1800mAh = 150g
  - 2700mAh = 250g

Flight times

- Hover:
  - 1800mAh : approx 11min
  - 2700mAh : approx 17min
- Agressive/stunt:
  - 1800mAh: approx 7 min
  - 2700mAh: aprox 13 min

Costs:

- FC: KK2.0 = 23 Euro
- Frame: F330 = 10 Euro
- ESC: 4x Turnigy Multistar = 4x 8 Euro = 32 Euro
- Motors: 4x 2822/17 1100kV = 4x 8 Euro = 32 Euro
- Battery: 2700mAh = 13 Euro
- Power-distribution: 5 Euro
- Propellors: 4x prop = 5 Euro
- Total: 120 Euro
- Receiver/Transmitter: 99 Euro
- Spares: Approx 20 Euro

U NAO 2

- Select the right FC for what you want
- Select the right TX/RX
- Select the right motor/prop for model size
- Select the right frame (or DIY!)
- Select the battery for you

CONGRATS!

NEXT workshop

- Date : To Be Determined
- Finalize design
- Plan ordering
- Place first orders
- Plan for next workshop when we DIY and FLY

Slide 135

This slide could have been yours

Thank you

- 136 slides
- Dozens of pictures
- Bad humor
- Useless info
- The winnar is you

Everyone Loves URLS

- http://wiki.techinc.nl/index.php/Justa_copter_1
- http://hobbyking.com
- http://3drobotics.com
- http://aeroquad.com
- http://ng.uavp.ch
- http://www.ecalc.ch