

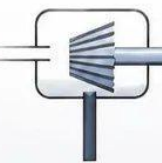


# Car physics

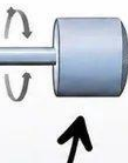
Automobiles, even

Electricity

**ENZO FERRARI**



Generator



**Ferrari  
LUC**



SABRINA CARPENTER DOES NOT KNOW HOW TO DIAGNOSE AND REPLACE A FAULTY CAMSHAFT POSITION SENSOR ON THE 2019 AUDI A4 2.0T, REQUIRING ACCURATE INTERPRETATION OF TORQUE SIGNAL OUTPUTS VIA OBD-II DIAGNOSTICS TO RESTORE ENGINE TIMING ACCURACY.



# What is a car

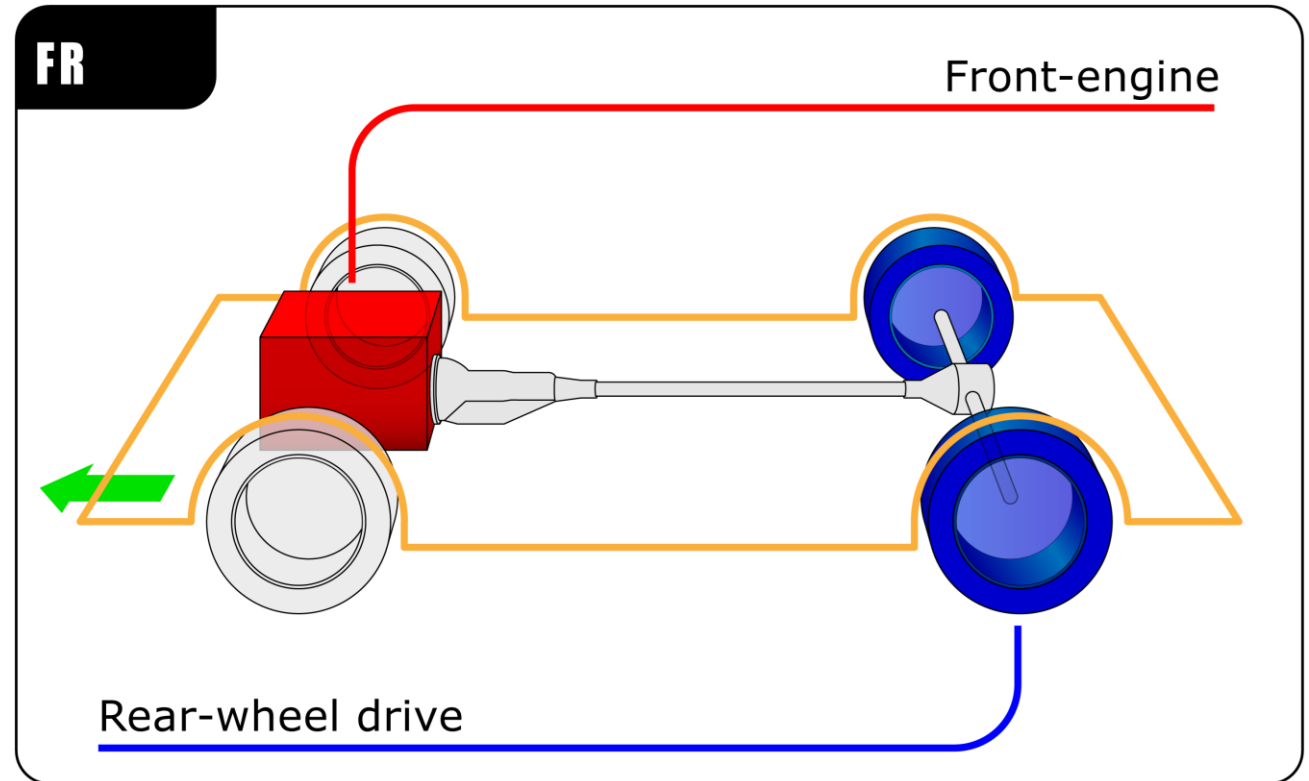
A vehicle with:

- 3 or more wheels
- That can move under its own power
- Has a suspension

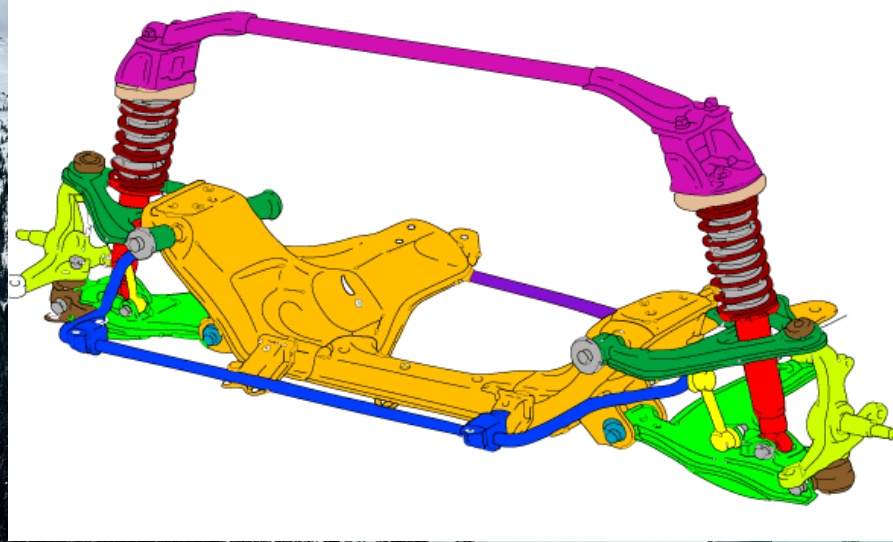


# How does a car's drivetrain typically work?

- Engine generates torque
- Clutch transmits torque to the gearbox
- Gearbox multiplies or divides the RPM and torque
- Differential receives torque and sends it to the wheels
- Tires push the car
- Tires on the ground also resist motion so the rest of the system gets a "reaction" torque back.







shocks  
springs  
subframe  
sway bar  
sway bar end link  
subframe brace  
shock tower brace  
upper shock mount  
alignment cam  
lower control arm  
upper control arm  
ball joint  
upright/knuckle

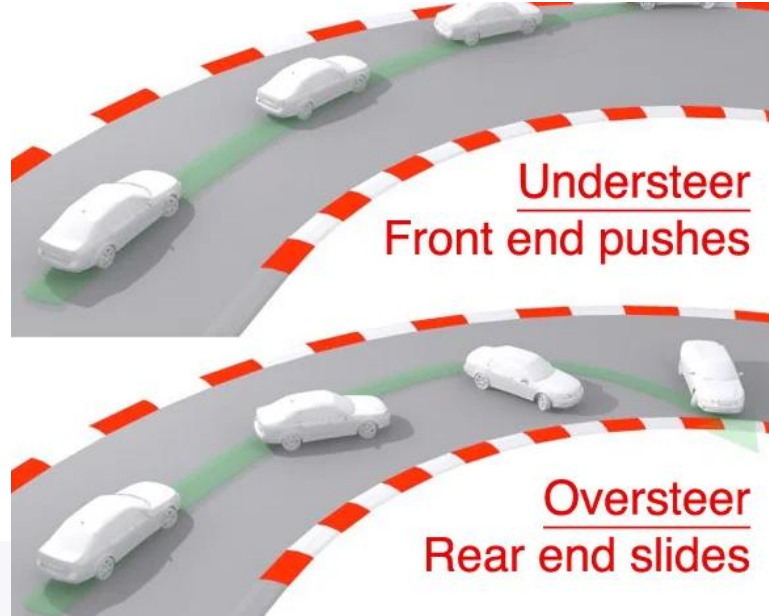
# How does a car's suspension work?

- The objective of the suspension is to keep the tires in contact with the ground.
- When you brake, the nose "dives" which loads the front wheels and gives you more grip
- This is what's known as weight transfer, and it's the fundamental way racing drivers control how a car moves



# Car handling 101

- Depending on the vehicle layout, weight distribution and driver skills a car can understeer, oversteer or be neutral.
- Understeer is when the car rotates less than what you input.
- Oversteer is when it rotates more than what you input.
- Neutral cars exist but they are mega rare.
- Some particularly terrible cars even have understeer and oversteer in the same corner (Ferrari F14 T).



POV: the last thing you see after deciding to take your Miata on a scenic canyon drive:





# Car handling 102

Cars CAN have handling tendencies, but they can also be influenced by the driver.

For example, in a car with the engine at the rear if you add throttle while cornering the car's front wheels lift.

This causes the rear to have more grip while the front gets less, this makes steering less effective, and you get understeer.

Anonymous 11/20/17(Mon)02:48:46 No.18233173 ▶ >>18233590 >>18233660 >>18233708 >>18235950 >>18237338

>HURRR snap oversteer DURRR

Anyone saying this has never driven a MR2. I absolutely guarantee it. You watched RCR's videos and BOOOM your comment on snap oversteer just joined the shit list of worthless comments on the bullshit snap oversteer comments on /o/. It's all sad, very sad indeed.

Sure the possibility for snap oversteer is there but it still takes a combination of at least two of the following to happen: 25 year old suspension that is fucked in some way, shitty tires and the driver doing something retarded. The last one is always involved and the primary reason for it and the first two points can be easily fixed for good. Seriously with lowering springs or updated roll bar the car has perfectly balanced handling and if you ever manage to get into slide they are very controllable.



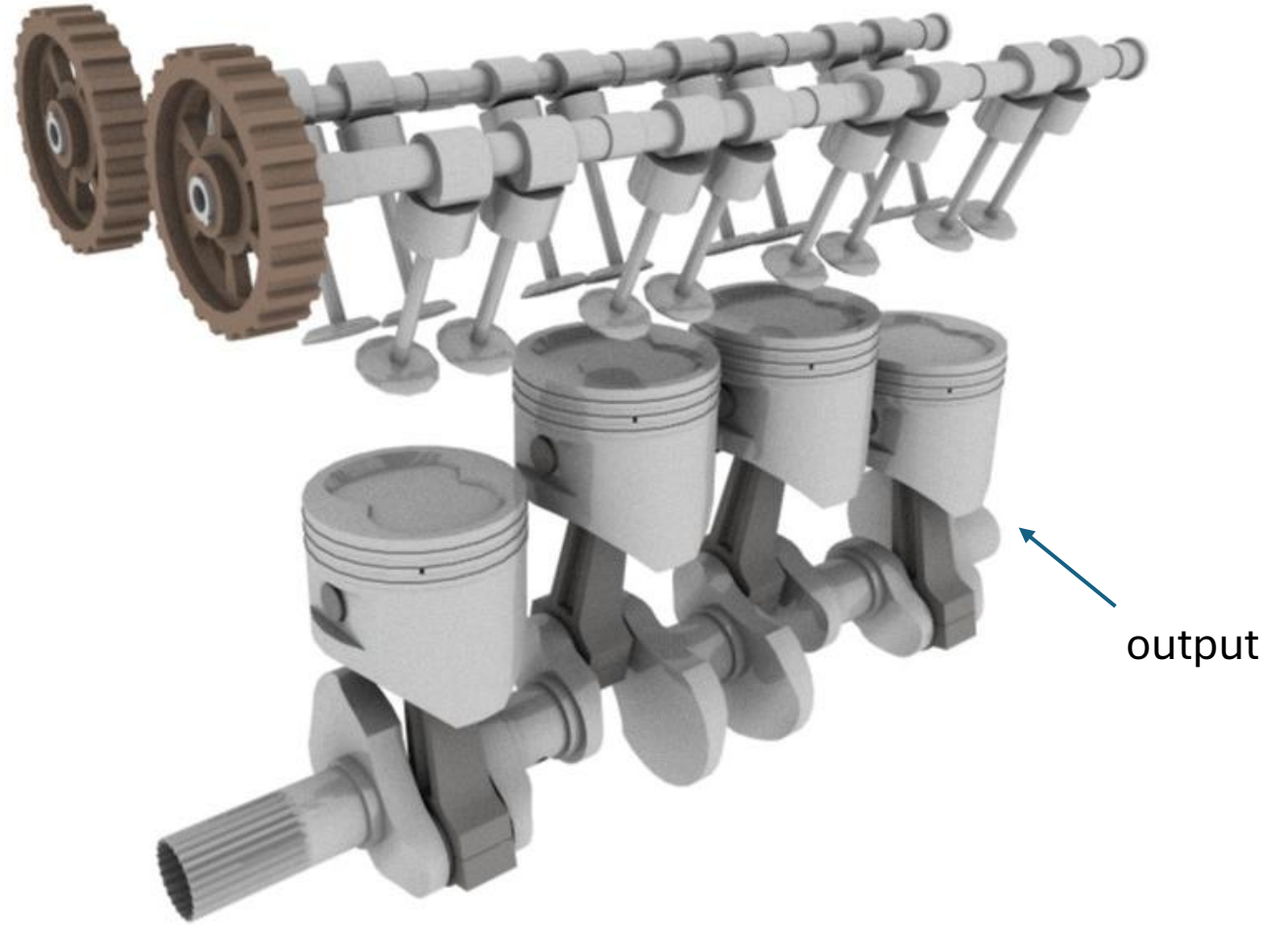
# Drivetrain parts

There's more than one,  
apparently

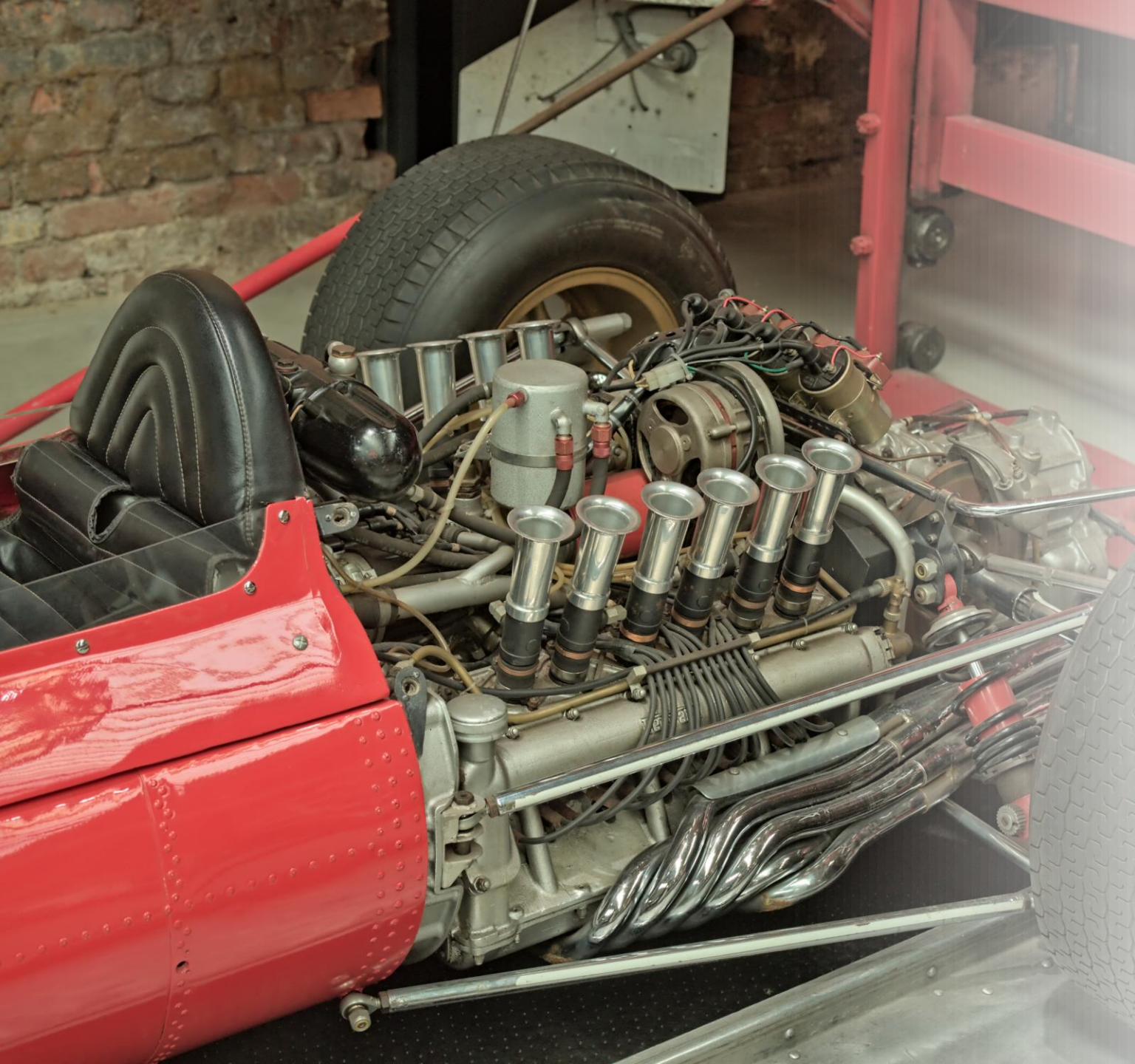


# Wot is an engine

- Fancy air pump
- Typically has an RPM limit (things would explode otherwise)
- You press the throttle pedal and the engine delivers torque
- It converts chemical energy to rotation
- Makes beautiful noises





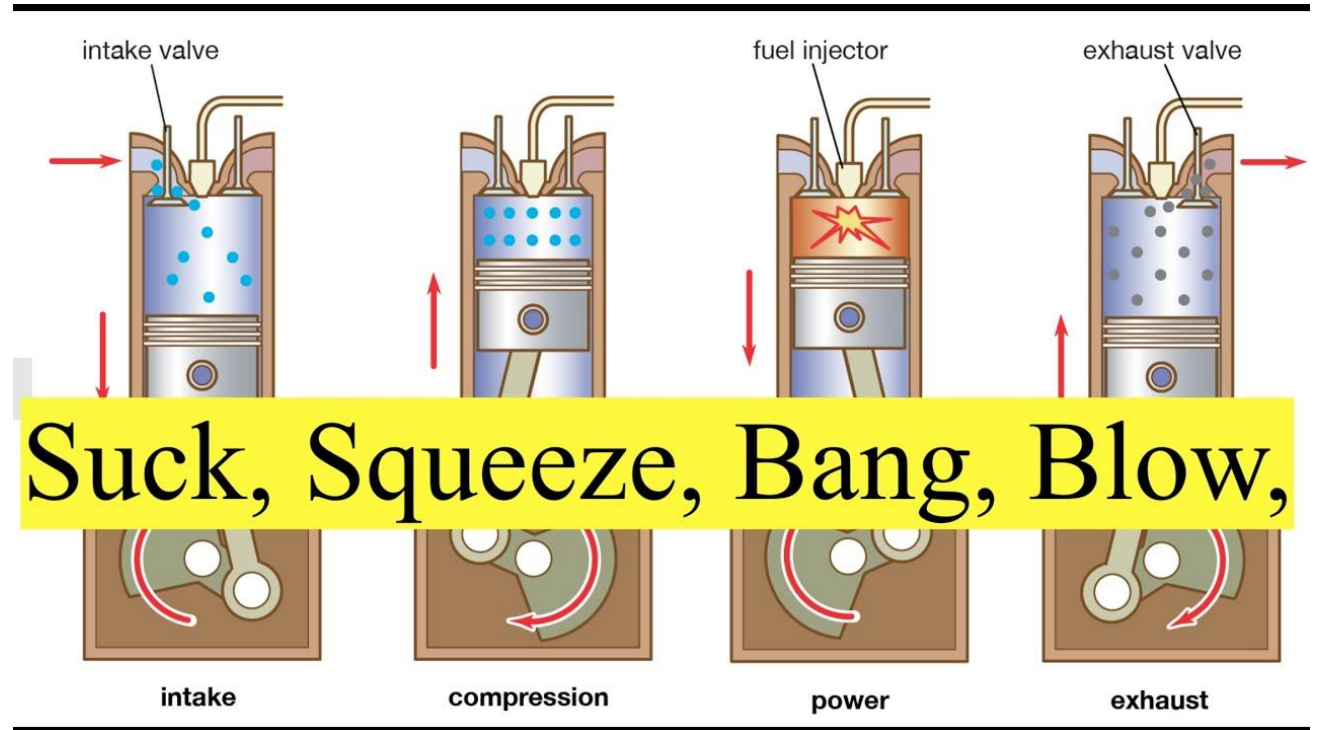


# How does an engine work (I)

- Engines mix air with fuel
- Gasoline engines use sparkplugs (Fancy arcing devices) to initiate combustion
- The amount of air that goes into the engine is typically regulated by a single butterfly valve, known as the throttle body.
- Depending on how much air is entering, the engine decides how much fuel to deliver.
- Engines will automatically open the butterfly valve to maintain idle.

# How does an engine work (II)

- Four stroke engines have four steps:
  - Suck: Air is drawn in by the piston moving down.
  - Squeeze: Air-Fuel mixture is compressed by the piston moving up.
  - Bang: Sparkplug ignites the mix and the explosion pushes the piston down.
  - Blow: The now combusted mix gets exhausted by the piston going up.





# How does an engine start?

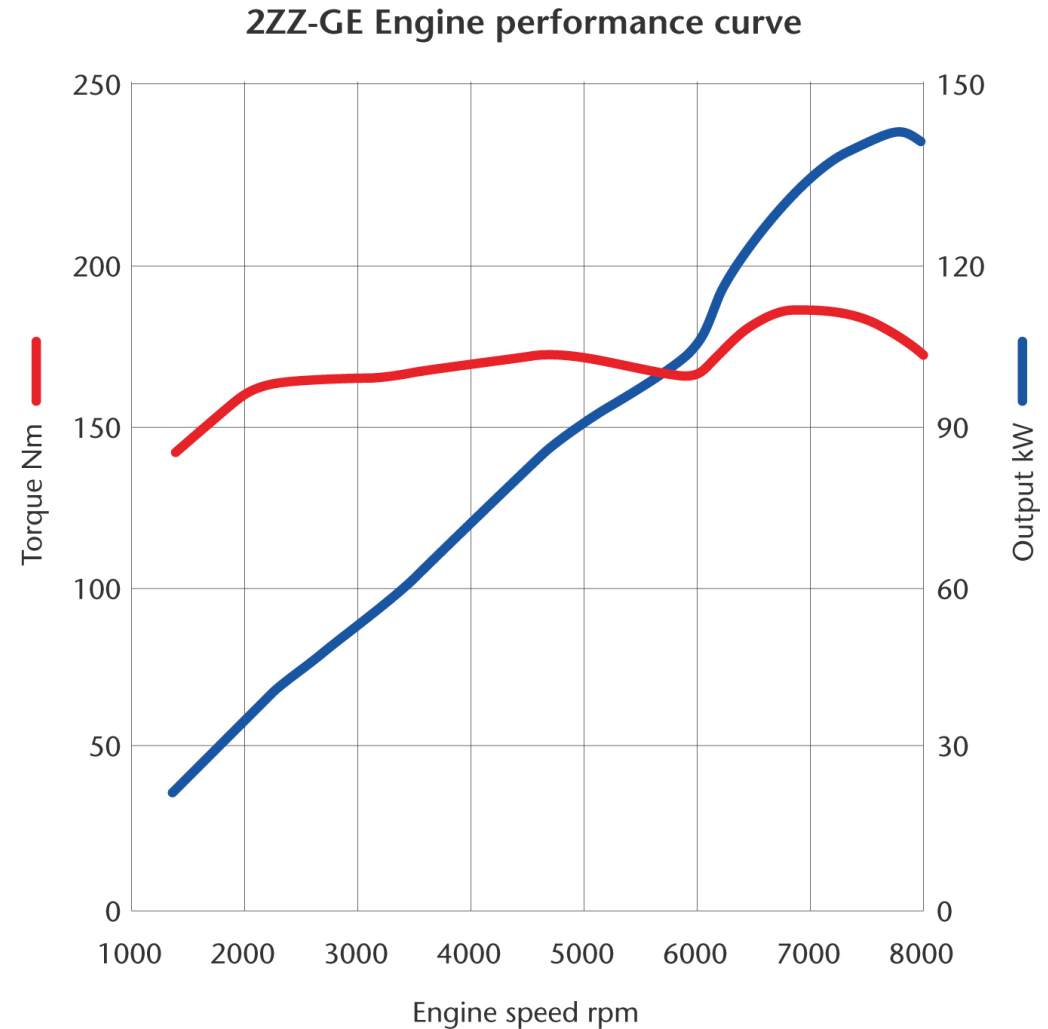
- Combustion engines cannot start on their own
- You need to somehow make them spin at a speed high enough for it to sustain combustion.
- You have some options:
  - Crank (could break your thumb)
  - External starter (used in F1)
  - Starter electric motor
  - Throwing a car down a hill and dumping the clutch

For some reason, people found the starter motor to be the most practical option and these days it's the most commonplace



# How do games simulate car engines?

- Engine is assumed to be a cylinder with a given inertia.
- Has a torque curve or map.
- Has a coast curve (when off throttle) that makes the car lose RPM when not given throttle.





# Gearbox

- Engines have an RPM limit
- We use a gearbox to make the wheels be able to run faster than the engine
- In most cars 4<sup>th</sup> gear is a 1:1 ratio (direct drive)
- This is what allows cars to run at higher speeds regardless of how high the engine revs
- (EVs typically have a single fixed gear for reasons)



# Clutch

- To be able to change gears we need to be able to disconnect the engine from the rest of the drivetrain.
- The clutch also acts to somewhat absorb sudden shocks.
- The clutch can be slipped to make taking off smoother.

**"Press the clutch, release it slowly and accelerate."**

**\*car engine turns off\***

**Me:**





# Differential

There are two driven wheels, but we have one engine output

There are three types of diffs:

- Open diff: Most common, splits torque evenly, makes the car a bit snappy on the limit
- Locked diff: Never sold in consumer cars, sometimes made by BMW owners by welding an open differential. Makes both wheels be physically connected together, makes drifting easier (but every other kind of driving miserable).
- LSD (Limited-slip): Used in performance or sporty cars, makes the car more predictable, best of both worlds.

## LSD:

**Car  
enthusiasts**

@initial.dreams

**Non car  
people**



# Chassis parts

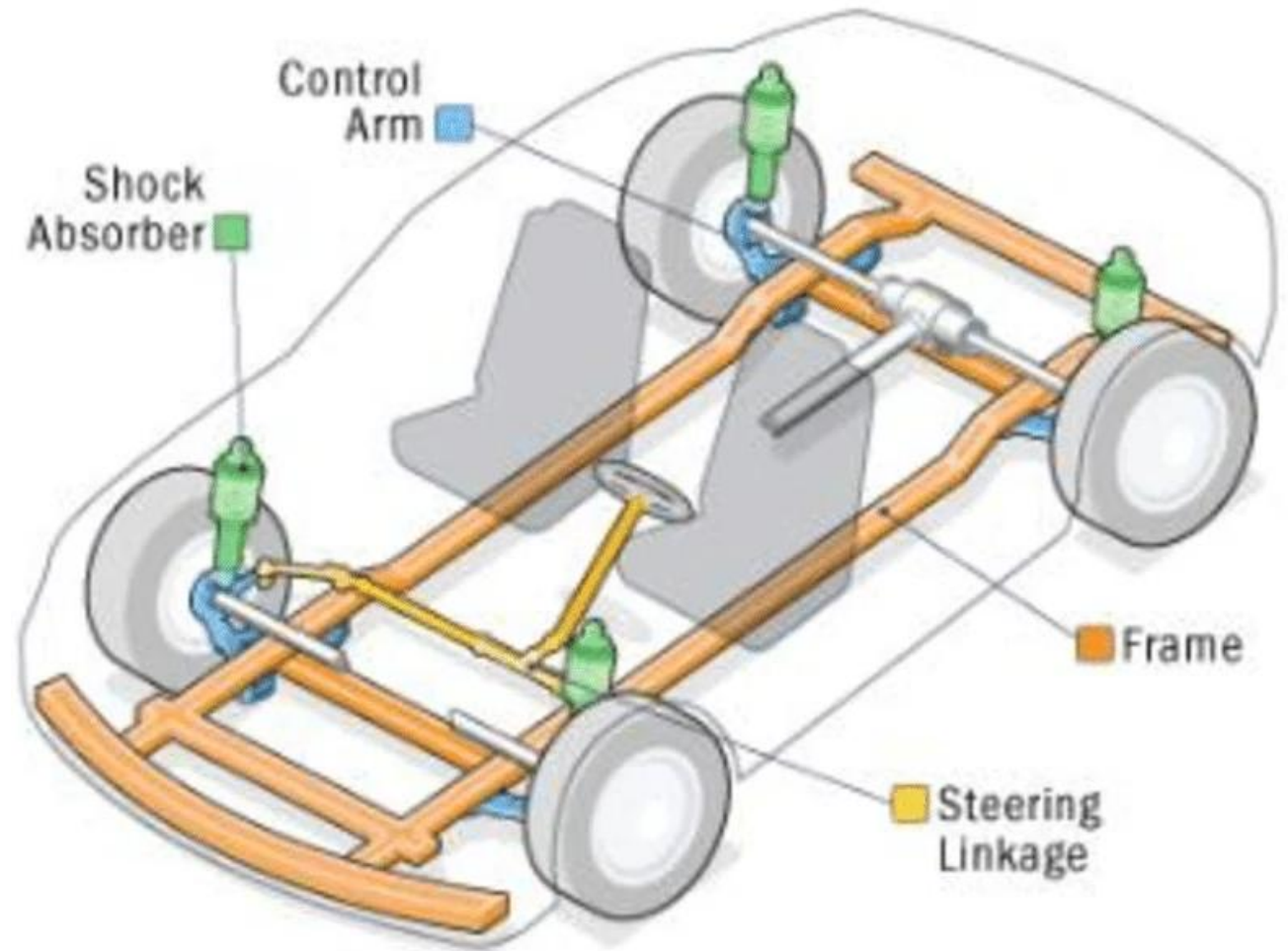
There's more to life than  
power





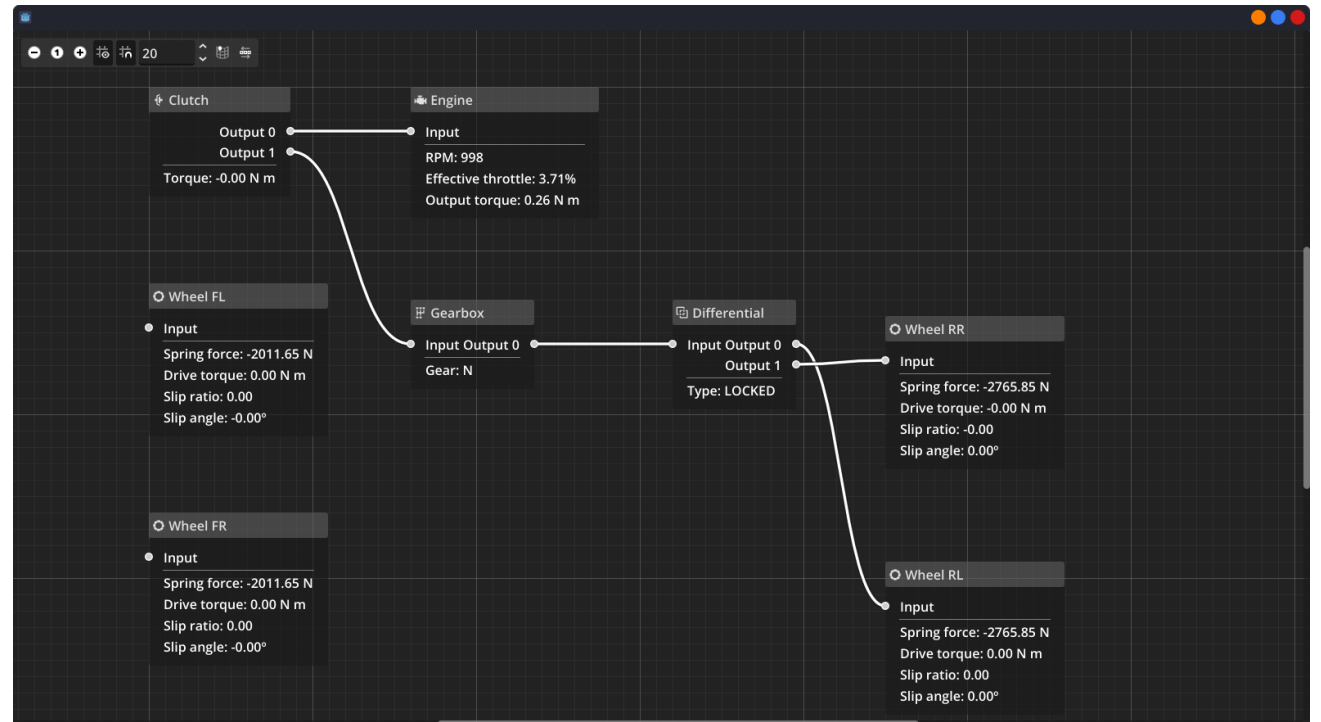
# Sprung and unsprung mass

- Typical consumer cars have a spring over damper system attached to some more or less complicated suspension geometry.
- The mass under the shock absorber (wheel, tire, hub, brake etc) is called **unsprung mass**.
- The mass above the shock absorber (the part that gets suspended) is called sprung mass.
- Generally extra unsprung mass is worse than sprung mass for handling.



# How I model the drivetrain

- I model the drivetrain as a tree of nodes.
- Each node corresponds to a real mechanical component.
- They can have one input and multiple output.



# Upstream and downstream

- First we update from the bottom (the wheels) and up
- Propagate upwards:
  - Angular velocity
  - Reflected inertia (inertia of all components on the way up added step by step).
  - Reaction torque
- Propagate downwards:
  - Reflected inertia
  - Torque

```
17  LNVehicleShaft::UpstreamData LNVehicleGearbox::get_upstream_data() {
18      const float ratio = drivetrain_settings->get_gear_ratio(current_gear);
19
20      LNVehicleShaft::UpstreamData downstream_data = get_child(0)->get_upstream_data();
21
22      if (ratio != 0.0f) {
23          return {
24              .inertia = (downstream_data.inertia / (ratio * ratio)) + drivetrain_settings->get_gearbox_inertia(),
25              .angular_velocity = downstream_data.angular_velocity * ratio,
26              .net_reaction_torque = downstream_data.net_reaction_torque / ratio
27          };
28      }
29      return {};
30  }
31
32  void LNVehicleGearbox::apply_downstream(const DownstreamData &p_data) {
33      const float ratio = drivetrain_settings->get_gear_ratio(current_gear);
34
35      float net_upstream_inertia = p_data.reflected_inertia + drivetrain_settings->get_gearbox_inertia();
36      get_child(0)->apply_downstream({
37          .torque = p_data.torque * ratio,
38          .reflected_inertia = net_upstream_inertia * (ratio * ratio)
39      });
40  }
41  }
```



# The crazy trick

- A big problem was figuring out in which order should things be processed
- I originally considered just grabbing all leaf nodes and going up the tree
- This didn't work very well
- The solution was to have a single node with no inputs, but two outputs be a "solver" and use push-pull.
- I choose the clutch to be the solver.

```
void LNVehicleClutchNode::update(float p_delta, const VehicleInputState &p_input_state) {
    LNVehicleShaft *drive_side = get_child(0);
    LNVehicleShaft *load_side = get_child(1);

    LNVehicleShaft::UpstreamData drive_side_data = drive_side->get_upstream_data();
    LNVehicleShaft::UpstreamData load_side_data = load_side->get_upstream_data();

    const float autoclutch_min = drivetrain_settings->get_autoclutch_min();
    const float autoclutch_max = drivetrain_settings->get_autoclutch_max();
    const float autoclutch_amount = 1.0f - CLAMP(
        Math::inverse_lerp(autoclutch_min, autoclutch_max, drive_side_data.angular_velocity * LNMath::AV_2_RPM),
        0.0f, 1.0f
    );

    const float clutch_input = MAX(p_input_state.clutch, autoclutch_amount);

    const float clutch_inertia = 0.0f;

    float drive_side_inertia = (0.5f * clutch_inertia + drive_side_data.inertia);
    float load_side_inertia = (0.5f * clutch_inertia + load_side_data.inertia);

    const float drive_momentum = drive_side_data.angular_velocity * drive_side_data.inertia;
    const float load_momentum = load_side_data.angular_velocity * load_side_data.inertia;
    const float total_momentum = drive_momentum + load_momentum;

    float full_lock_angular_velocity = total_momentum / (drive_side_inertia + load_side_inertia);
    float full_lock_torque = drive_side_inertia * (full_lock_angular_velocity - drive_side_data.angular_velocity) / p_delta;

    float limit = drivetrain_settings->get_clutch_max_torque() * (1.0f - clutch_input);
    current_torque = -CLAMP(full_lock_torque, -limit, limit);

    drive_side->apply_reaction({
        .torque = -current_torque,
        .reflected_inertia = clutch_inertia
    });
    load_side->apply_downstream({
        .torque = current_torque,
        .reflected_inertia = clutch_inertia
    });
}
```



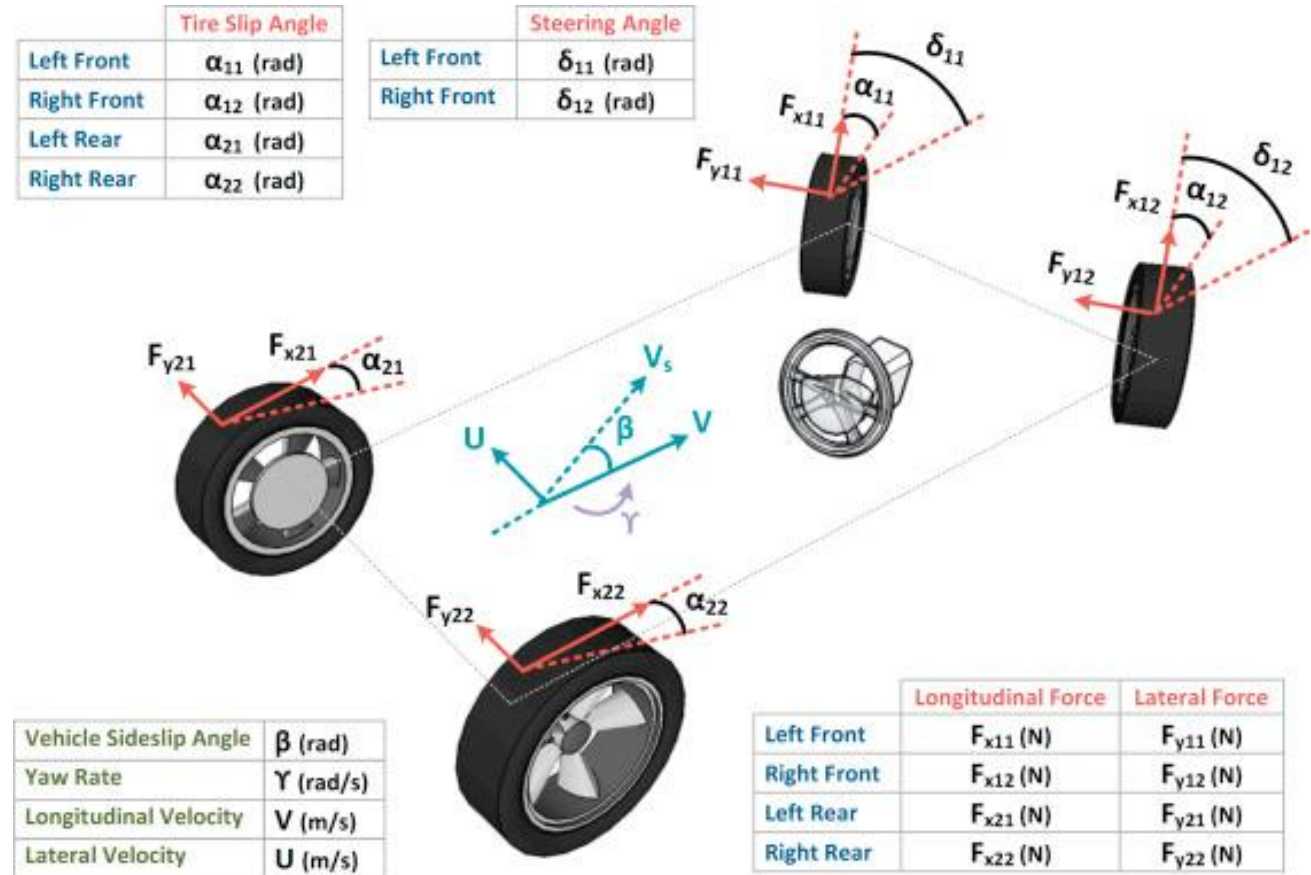
# Tyres

They are round



# How tyres work

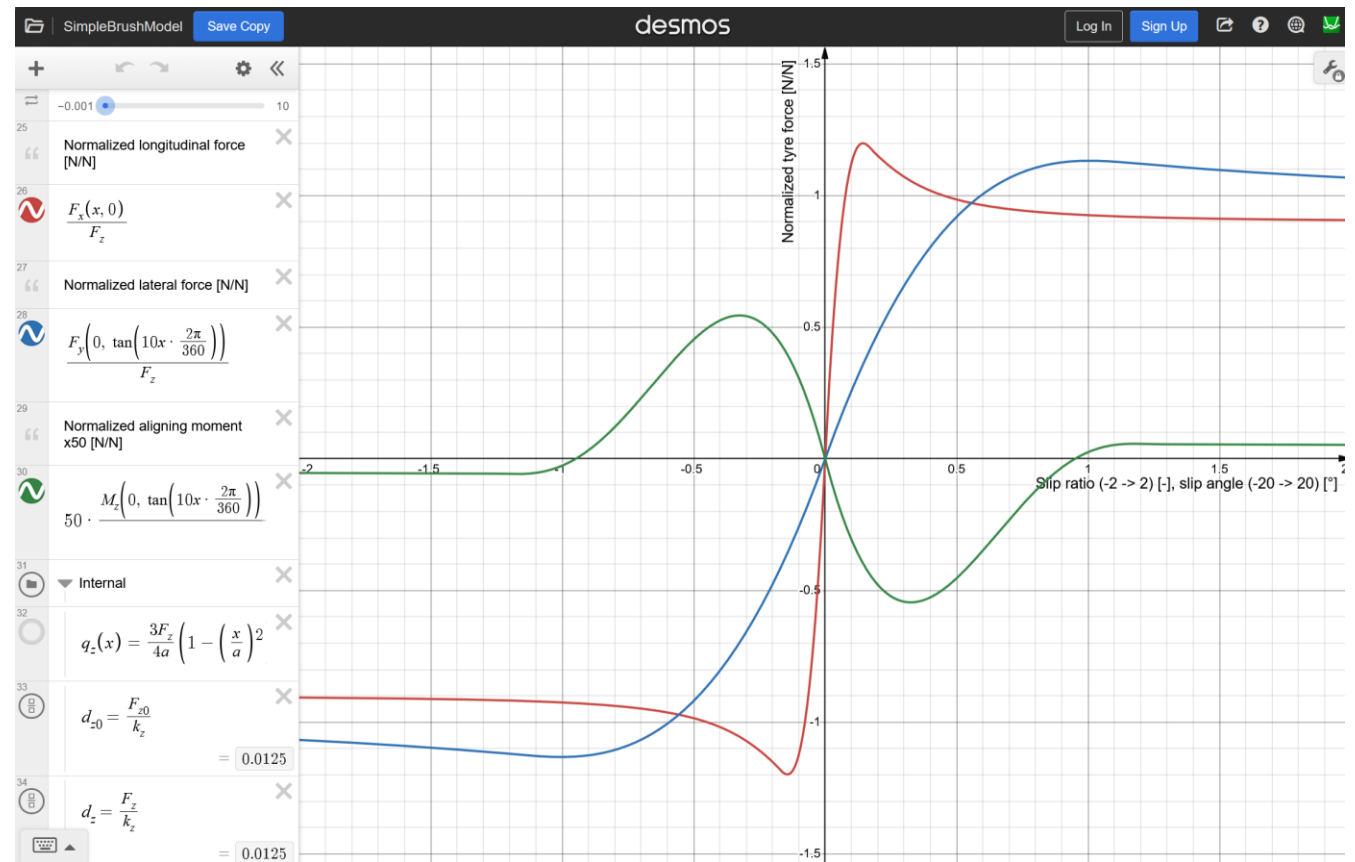
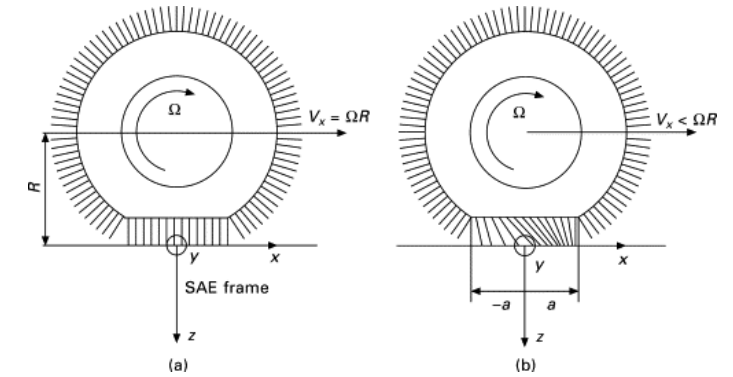
- Tyres only push when they are slipping.
- Slip is defined as two distinct values, the slip ratio and the slip angle.
- The slip ratio is the difference of the speed of the spinning tyre at the point where it touches the ground and the ground itself.
- The slip angle is the difference between the forward direction of the tire and the direction it's traveling in.





# How 2 simulate tyres

- Use a tyre model
- At minimum, they use the slip ratio, slip angle and the force being exerted by the suspension on the wheel.
- There are three major ways to do things.
  - Pacejka's magic formula, named because he made it up.
  - Brush, where the wheel is simulated as a bunch of bristles of a brush.
  - Lookup tables, where you grab a real tyre, put it through various levels of vertical force and slip ratio/angle and write it into a table.
- They are usually depicted using a graph
- They return longitudinal and lateral force in newtons

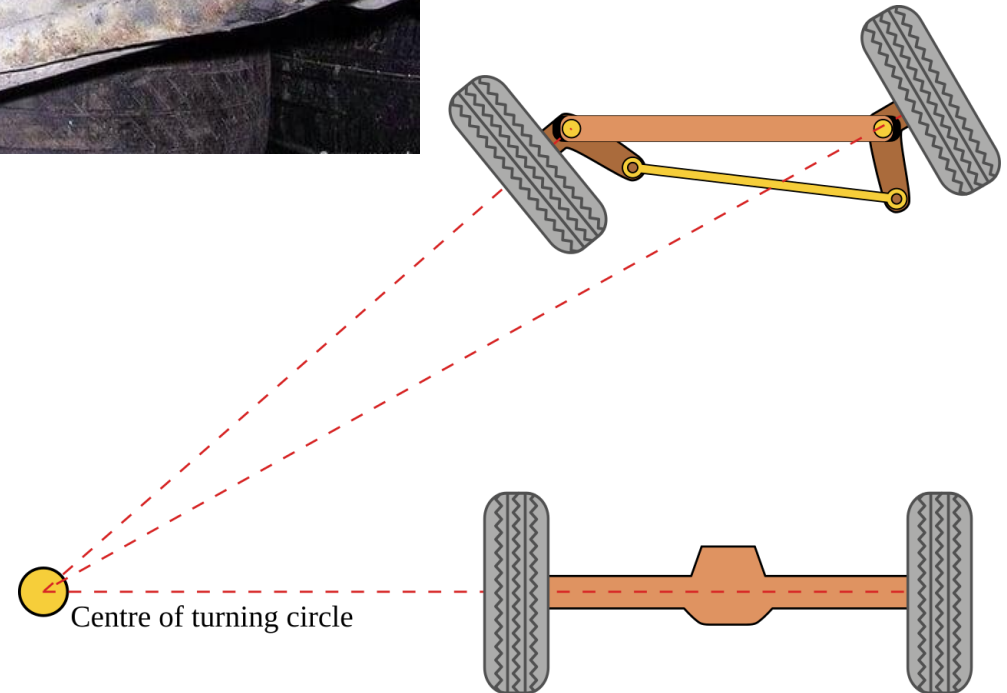


# There's a lot, a lot more

This was just a super bird's eye view of cars

There are a lot of things we haven't discussed:

- Throttle models
- Suspension geometry
- Camber/caster/toe
- Aerodynamics
- Anti-roll bars
- Slip ratio/angle calculation
- Turbos
- Sound simulation
- Ackerman steering





# Now you know how to build a car

Please use this knowledge responsibly



## Miata gang offers you choccy milk



## They shall return next Wednesday

