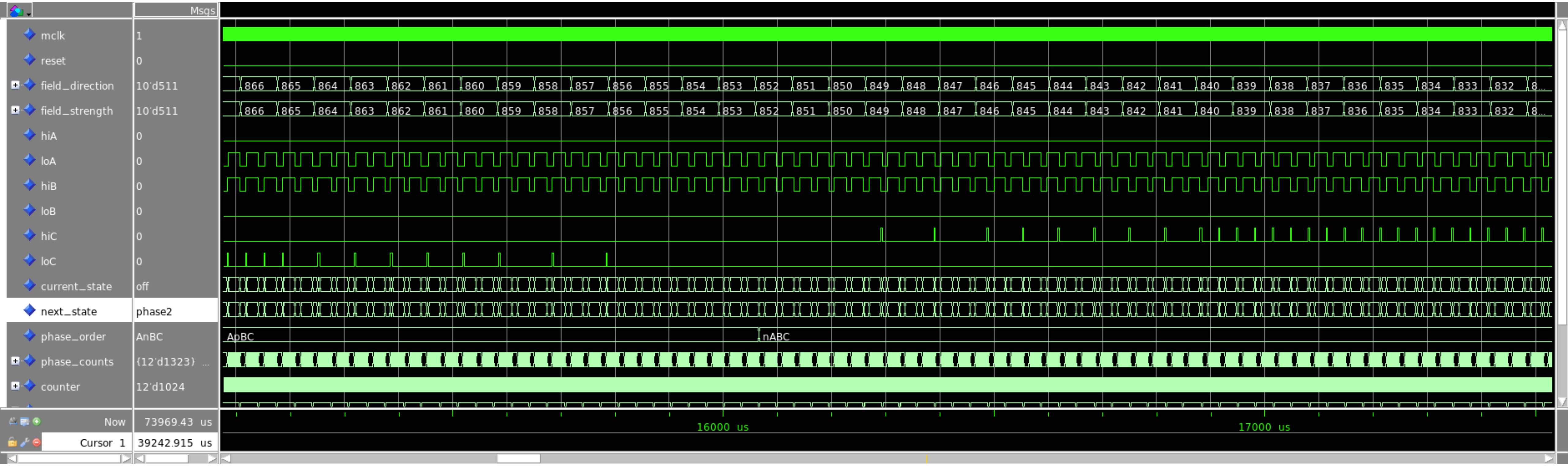
Faculty of Mathematics and Natural Sciences

FPGA motor control, why and how?



UNIVERSITY OF OSLO

Yngve Hafting
Cand.Scient
Department of Informatics,
Robotics & Intelligent systems group

2022-01-03

Overview

Part 1 : High level overview

- Why motor control?
- Why FPGA?
- General motor driver system

Part 2: Motor Control basics

- commutation of three phase motors
- Sensorless principles
- Field oriented control
- Slow speed challenges with FOC

Part 3: Output control

- Pulse modulation techniques
  - Sinusoidal PWM
  - Space vector modulation
- Challenges with SPWM and SVPWM
- Modulation summary

Part 4: Motor control algorithm status

- What is left to explore

Why

Stepping into the real world your robot should come with

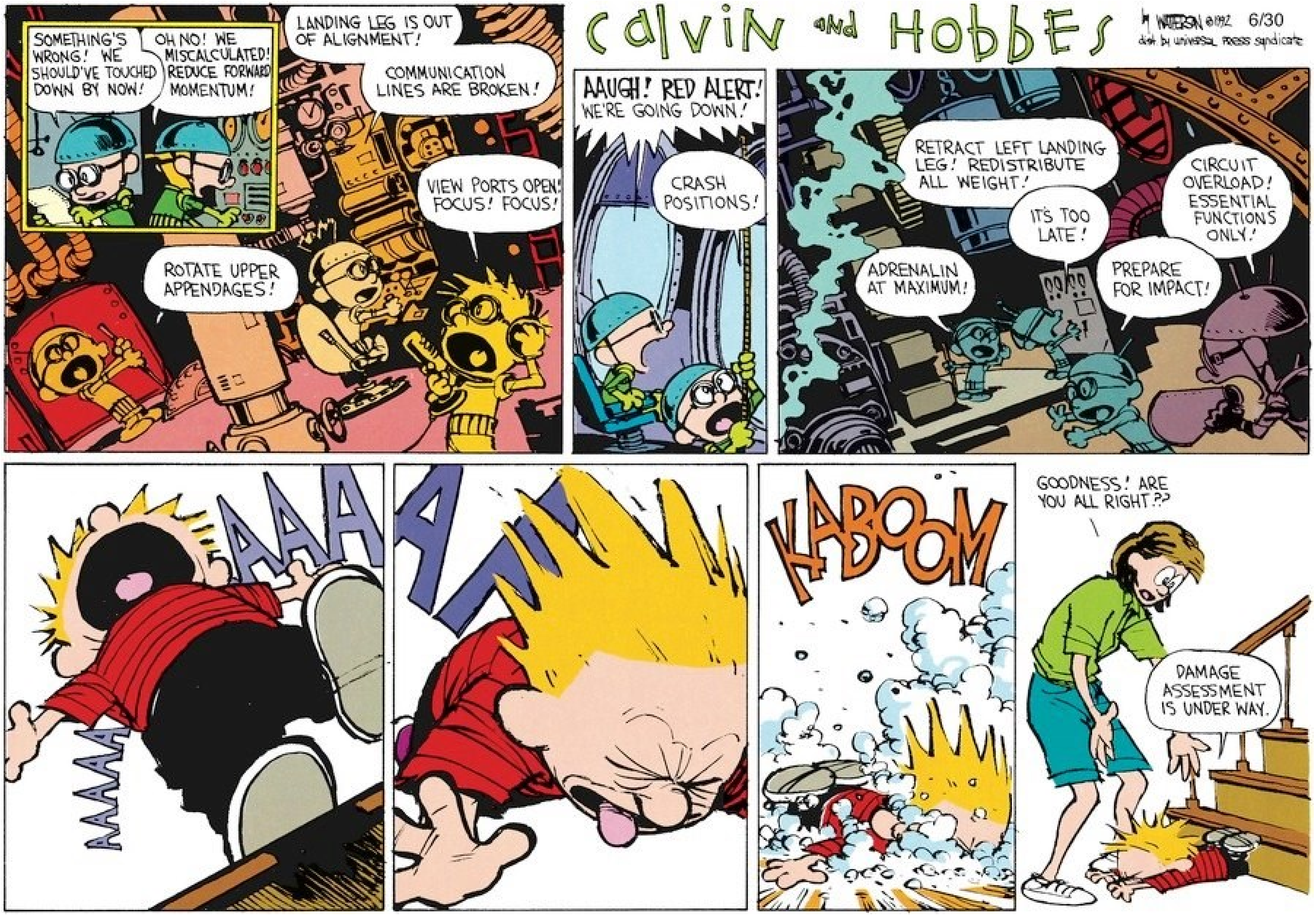
Quick reflexes:

- Low level control that
  - react swiftly
  - respond when parameters cannot be met
    - Accurate readout
- Fast link between high and low level control
  - Few steps
    - not a multi-branch tree
  - Few wires
    - Easier to build & repair

Lasting batteries

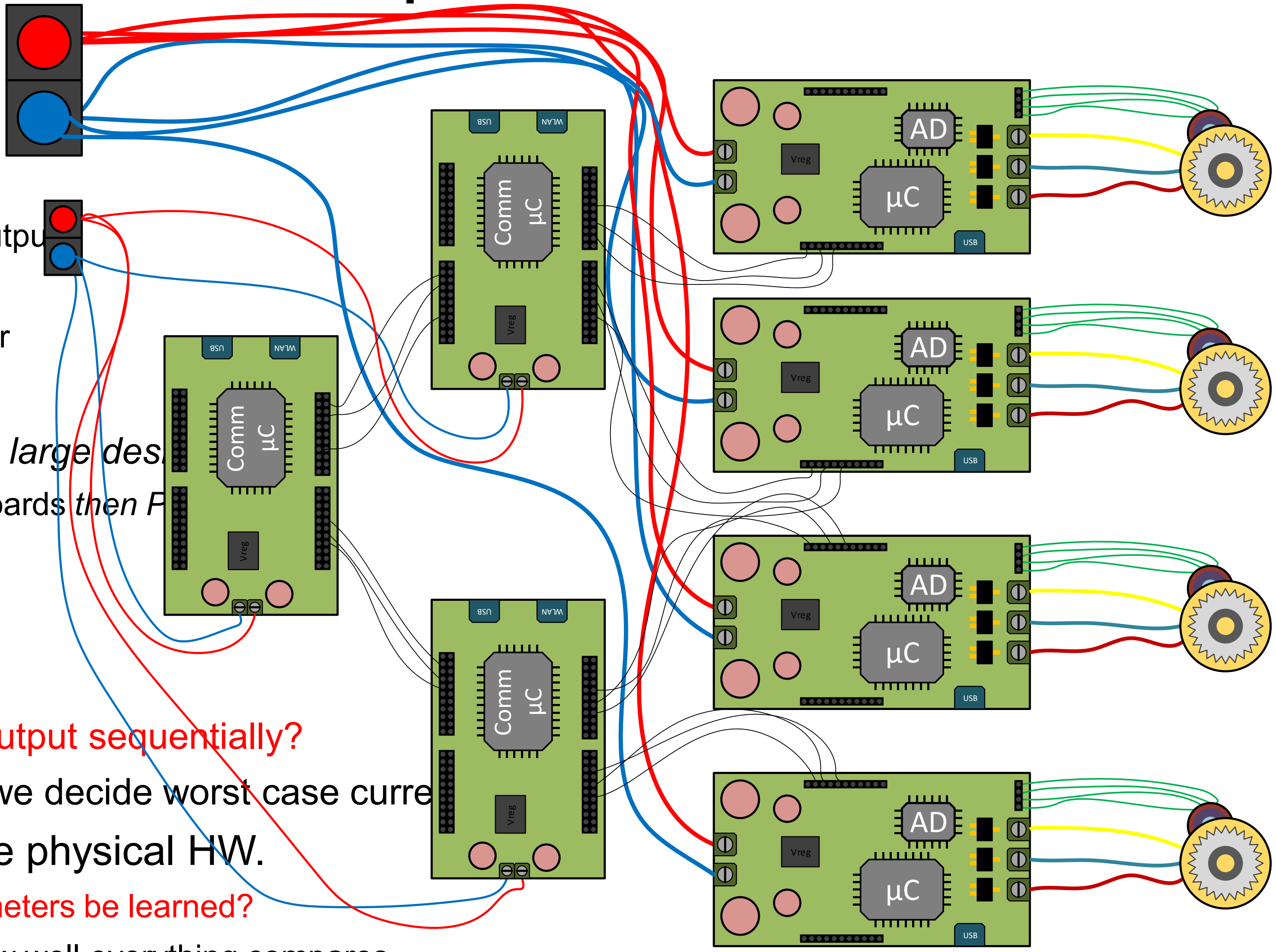
- Efficient controllers
  - Light weight body

etc



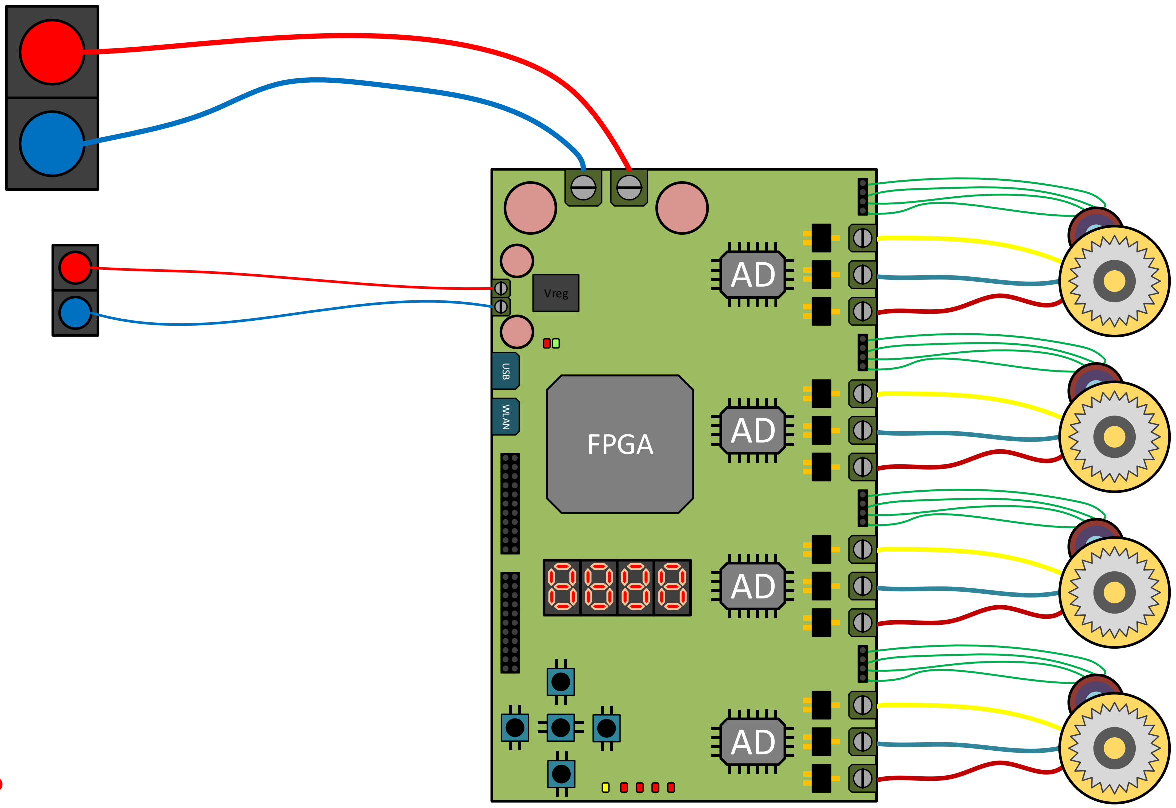
• Why FPGA system for control? µController can do all...

- IO galore:
  - Control N motor output (achievable -> 10+)
  - Sensor input not fixed
    - Input should synchronize with PWM outpu
      - *has to be done at a low level.*
    - Input could be more than one AD/motor
- Wire chaos
  - *Reduce the number of circuit boards in large desi*
    - *12-DOF ODR Robot: 6* µC, 3 comm-boards *then F*
- *Faster and more efficient control loops*
- Current control
  - If voltage is high enough:
    - Does it make sense to do PWM output sequentially?
    - Power design is easier, because we decide worst case curre
- Test different algorithms using the same physical HW.
  - To what degree can and should motor parameters be learned?
  - We know that ML *has* been used, but not how well everything compares.



• Why FPGA system for control? μController can do all...

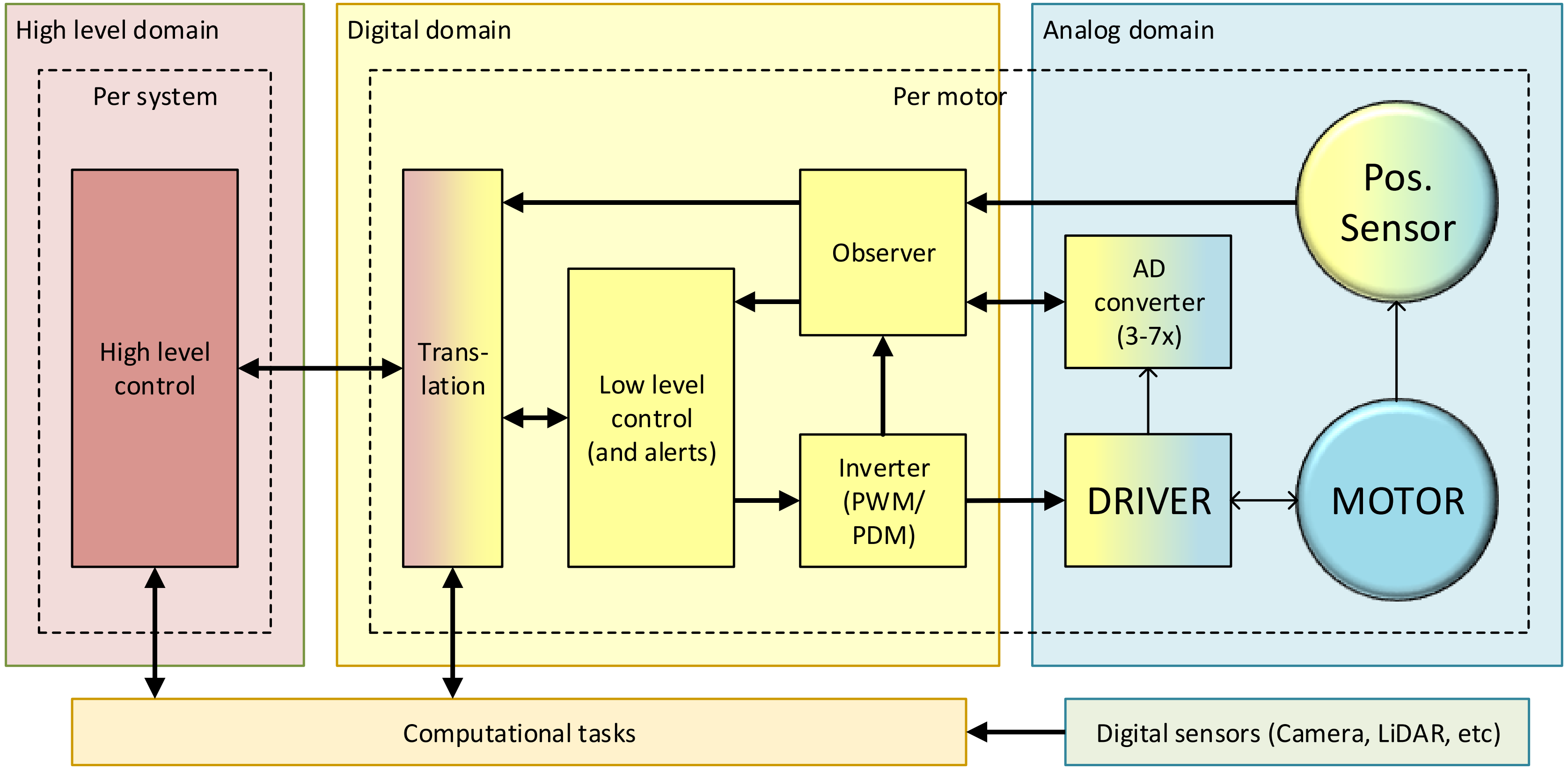
- IO galore:
  - Control N motor output (achievable -> 10+)
  - Sensor input not fixed
    - Input should synchronize with PWM output
      - *has to be done at a low level.*
    - Input could be more than one AD/motor
- Wire chaos
  - *Reduce the number of circuit boards in large designs*
    - *12-DOF ODR Robot: 6* μC, 3 comm-boards *then PC*
- *Faster and more efficient control loops*
- Current control
  - If voltage is high enough:
    - Does it make sense to do PWM output sequentially?
    - Power design is easier, because we decide worst case current
- Test different algorithms using the same physical HW.
  - To what degree can and should motor parameters be learned?
  - We know that ML *has* been used, but not how well everything compares.



UNIVERSITY
OF OSLO

General controller => What to do with it..

- All tasks in the digital and high level domain *can* be done in an FPGA.
- Creating everything all at once is too much
- Tasks
  - Create an inverter that can
    - talk to the observer
    - test all PWM algorithms at any rate.
    - be reprogrammed
  - Create observer that can
    - Utilize different AD and sensor combinations
    - Use AI/ML or Heuristics (LUTs) to provide motor state
  - Create a system that can be used to demonstrate benefits and pitfalls by using FPGA for Robot control.



Part 2 Motor control basics

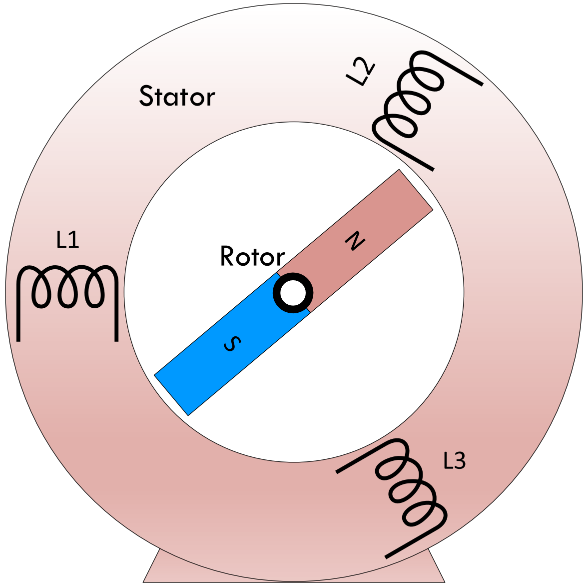
- Principles for the 3 phase motor
- Possible states in motor control
- Sensorless measurements
- Field oriented control

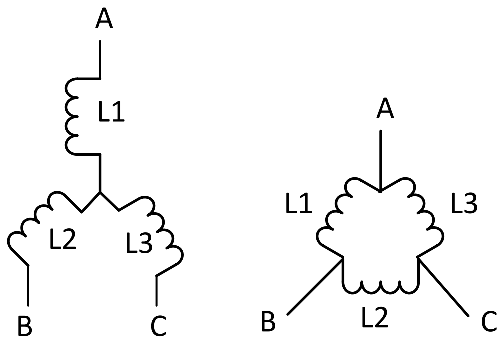
- Challenges at slow speed

3 phase motor principles

- Motor principle
  - (more poles)..
- 3 phase motor diagram
  - Wye is «considered» easiest to explain,
    - Texts often stop there...
- Inverter drive circuit
  - 6 transistors to apply PWM to:
  - Outputs are either high, low or disabled
    - Hi+ Lo + (Lo)
    - Lo + Hi + (Hi)

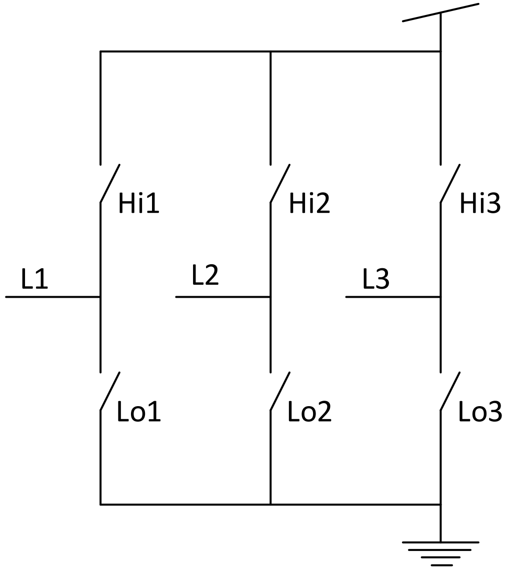
Motor



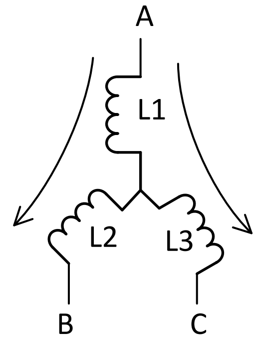


Y «Wye» and «Delta»

not connected at all = stepper motor

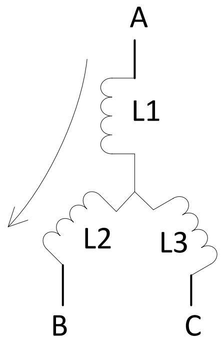


Wye vs Delta (no center tap)

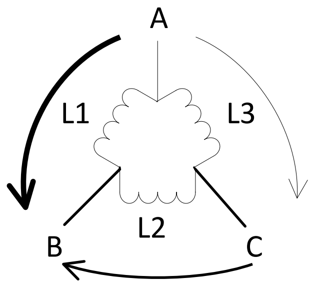


A High, B low, (C is off)

- Wye:
  - Current flows through L1 and L2
  - Current is $\frac{V}{2R_L}$
    - I_{L2} is $-\frac{V}{2R_L}$ if we use to-center as positive direction

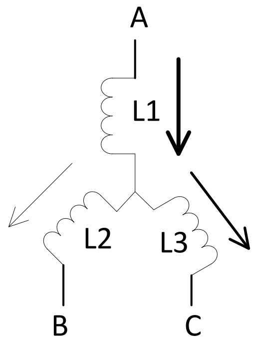


- Delta
  - Currents flows through all solenoids:
    - $I_{L1} = I_{AB} = \frac{V}{R_L}$
    - $I_{L2} = I_{BC} = -\frac{V}{2R_L}$
    - $I_{L3} = I_{CA} = -\frac{V}{2R_L}$

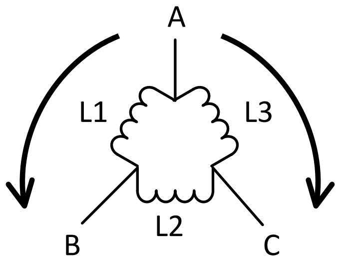


A high B and C low

- Wye
  - Current flows through all solenoids
    - $I_{L1} = \frac{V}{1{,}5R_L} = \frac{2V}{3R_L}$
    - $I_{L2} = -\frac{I_{L1}}{2} = -\frac{V}{3R_L}$
    - $I_{L3} = -\frac{I_{L1}}{2} = -\frac{V}{3R_L}$

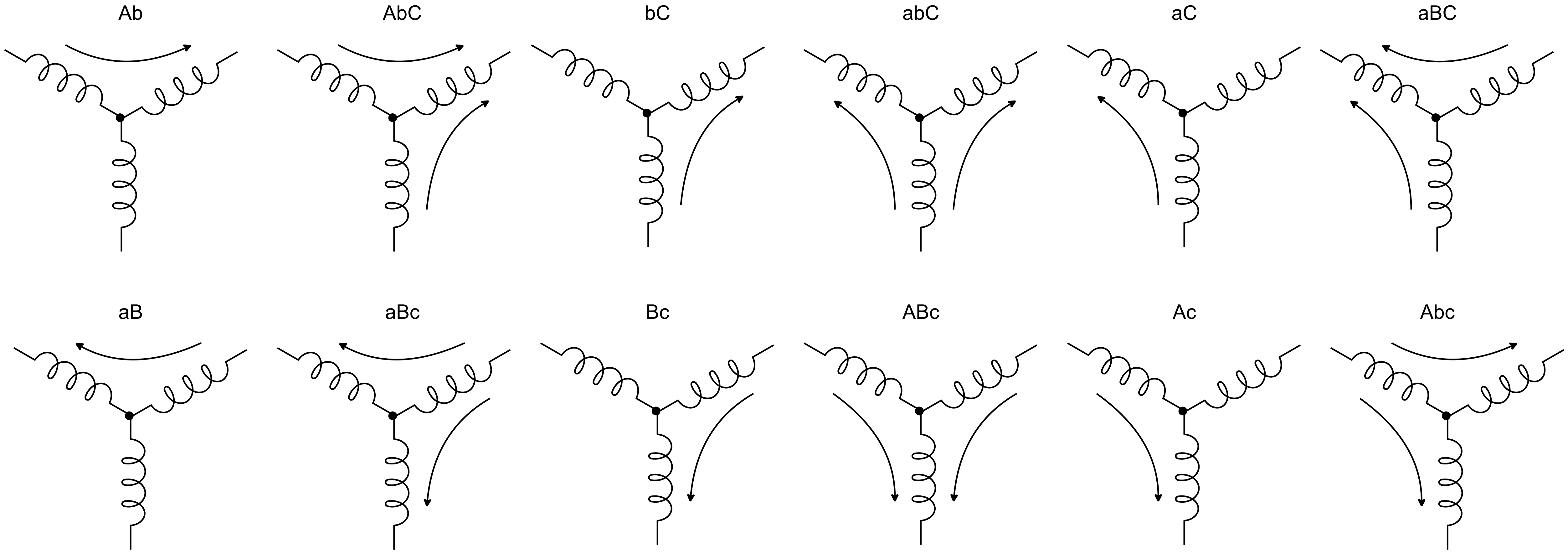


- Delta
  - $I_{L1} = \frac{V}{R_L}$
  - $I_{L2} = 0$
  - $I_{L3} = -\frac{V}{R_L}$



States for motor control

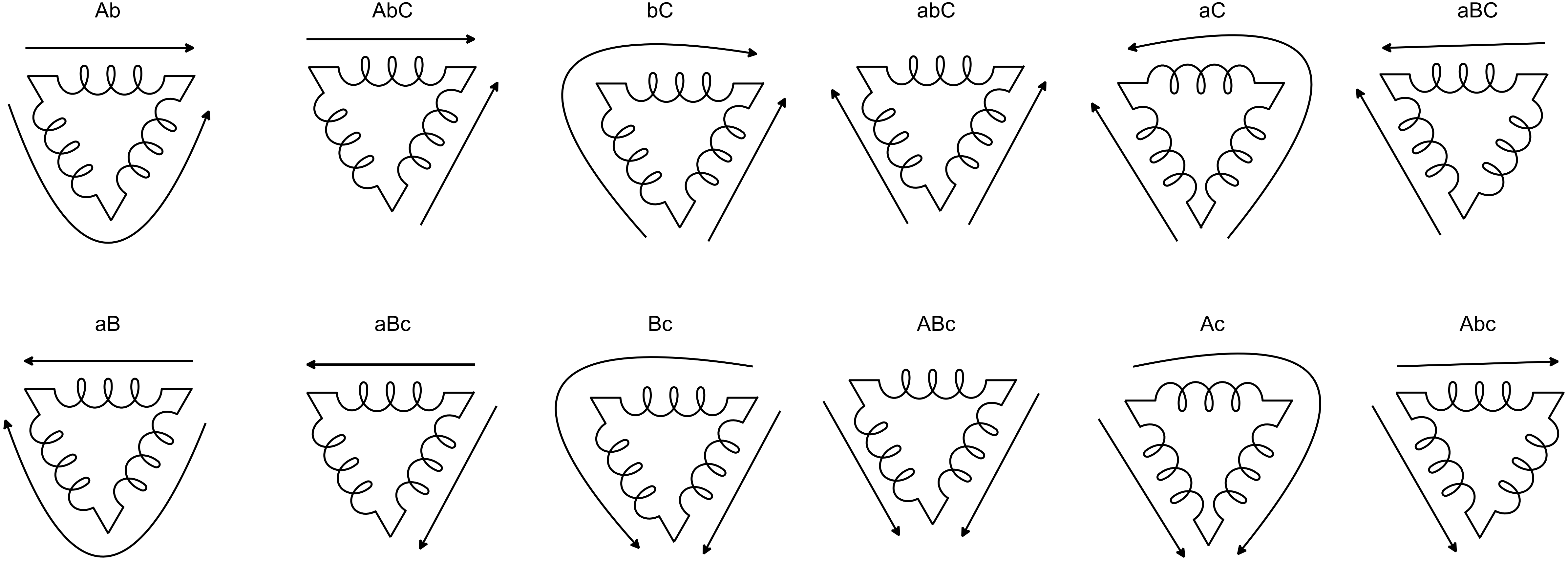
Wye connection, All current states



States for motor control

Delta-connection, all 12 states

Note:



Commutation scheme

- Ab
- AbC
- bC
- abC
- aC
- aBC
- aB
- aBc
- Bc
- ABc
- Ac
- Abc

In this scheme…

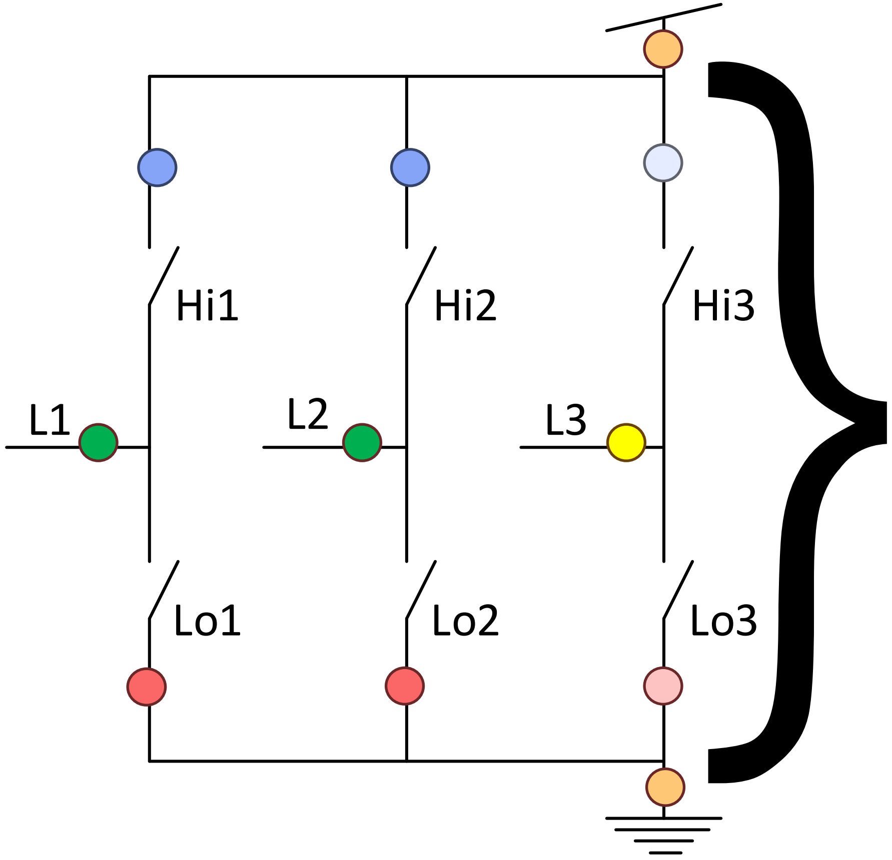
- Wye motor will lead by 30 degrees on a Delta motor, both turning same direction
  - Ex
    - Wye has
      - 2 coils active at Ab
      - 3 coils active at AbC
      - …
    - Delta has
      - 3 coils active at Ab
      - 2 coils active at Abc
        - Same as Ab for Wye

Sensors and sensorless

- "Sensor" based systems generally use a rotational or positional encoder (Quadrature or absolute encoder)

- Sensorless systems use a number of AD converters, that can be found on various locations in the inverter driver circuitry
  - Measure Voltages across
    - Shunt resistors for output
      - Two or three phases
    - Low or high side shunt resistor
      - for two or three phases
    - Low or high side common shunt resistor
    - Current supply voltage

  - Measurement can be performed
    - *When pulsing*
      - To help determine power and torque output
    - *When not pulsing*
      - To find back EMF to determine position and
      - To help determine power and torque
    - *During transitions* is not a good idea (synchronize!)



Field oriented control (FOC)

- The goal is either to control motor
  - speed
  - position
  - torque
  - acceleration (traction control)

- *FOC is achieved if we can set strength and direction of the magnetic field in the motor.*

- Vector control using Clarke Park
  - Uses AD-measurements to find the strength and direction of the motor currents
    - *Back EMF measurement is only possible at speed*
  - Transforms vectors to radial and tangential values
  - Input desired tangential values to achieve torque (current)
  - Use PID control to make sure radial value is 0 and tangential output becomes as desired
  - Transform back to get output values

Clark-Park transformation and PI regulation

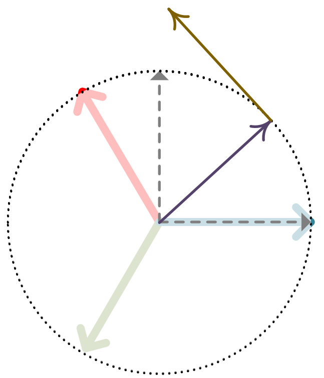
1: Transform all signals to xy plane (=> We go from 3 signals to 2)

2: Transform to radial/ tangent plane

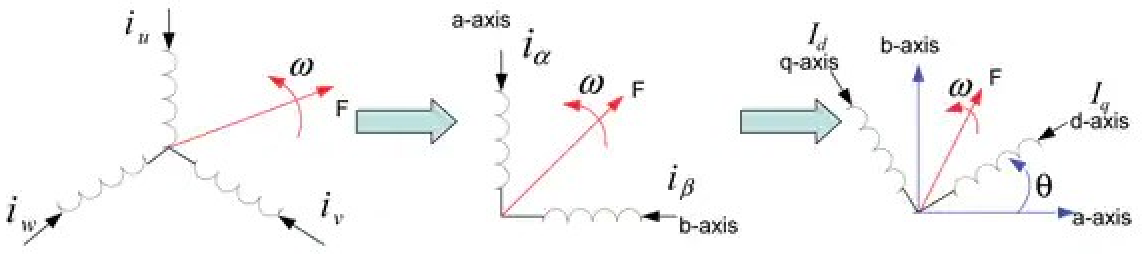
3: PI regulate so that radial component is 0 and tangential current is proportional to the motor torque.

4: Reverse Transform back to xy

5: Reverse transform back to three phase to get PWM timing.



- Makes torque calculations easy.
- PI will smooth out any imperfections since we use current feedback
  - *Reactive, not predictive…*



Clarke Transformation

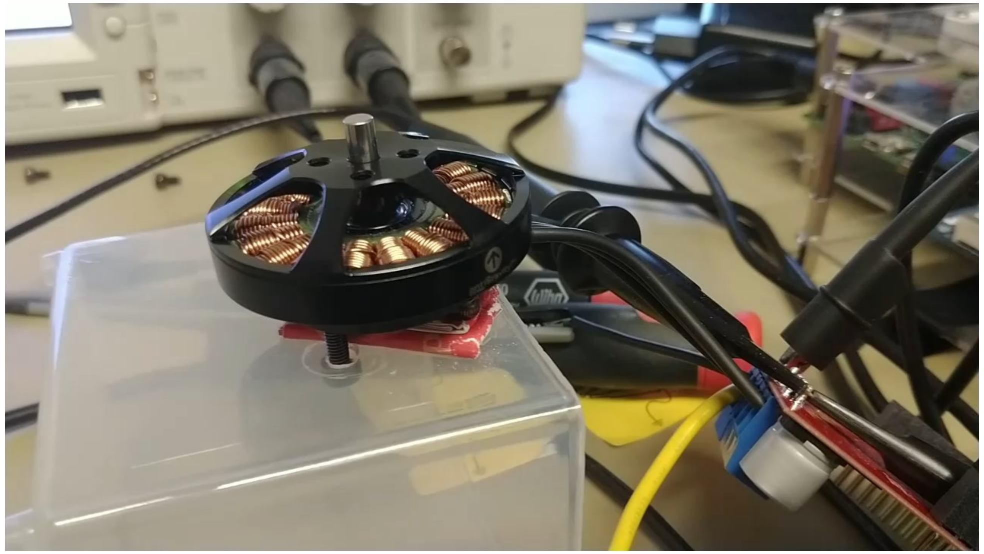
$$\begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix} = \begin{bmatrix} 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix}$$

Park Transformation

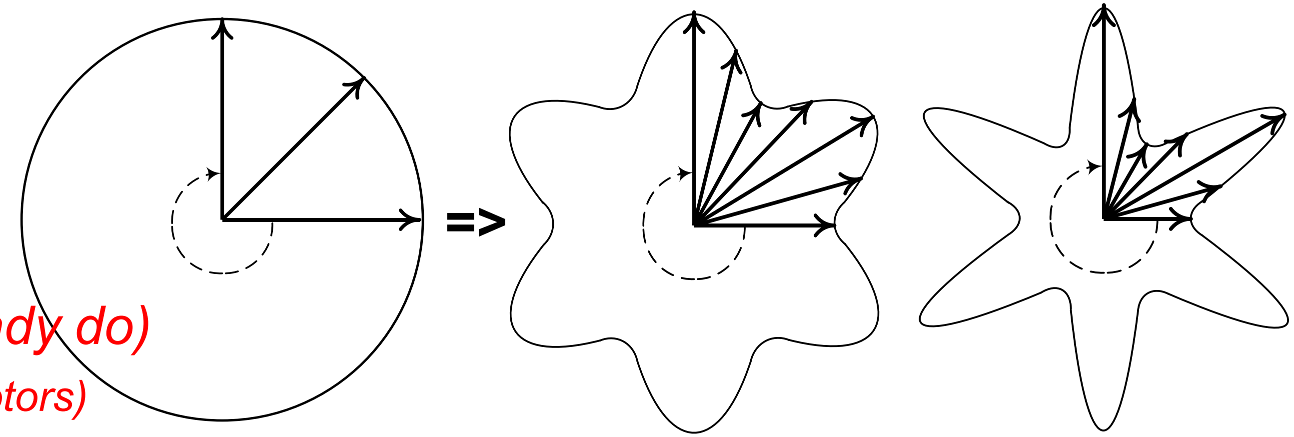
$$\begin{bmatrix} I_d \\ I_q \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} i_\alpha \\ i_\beta \end{bmatrix}$$

Is it even possible to «rotate» a field in stator?

- Feeding a «perfect circle» (PWM' based)
  - Some «field-directions» are weak
    - Others are much stronger
    - **Independent of motor position**
      - *you can feel this…*
    - Current draw is small when weak
      - *Motor acting as a transformer?*
      - ***Cogging Torque..?***



- Will toggling between strong vectors *always be a better option?*
  - *At what rate..? (since this is what we already do)*
    - Adding noise *seems to be used (ODR, Zero motors)*
    - *Addressed in paper «Cogging Torque Reduction in Brushless Motors by a Nonlinear Control Technique», P. Dini and S.Saponara 2019.*

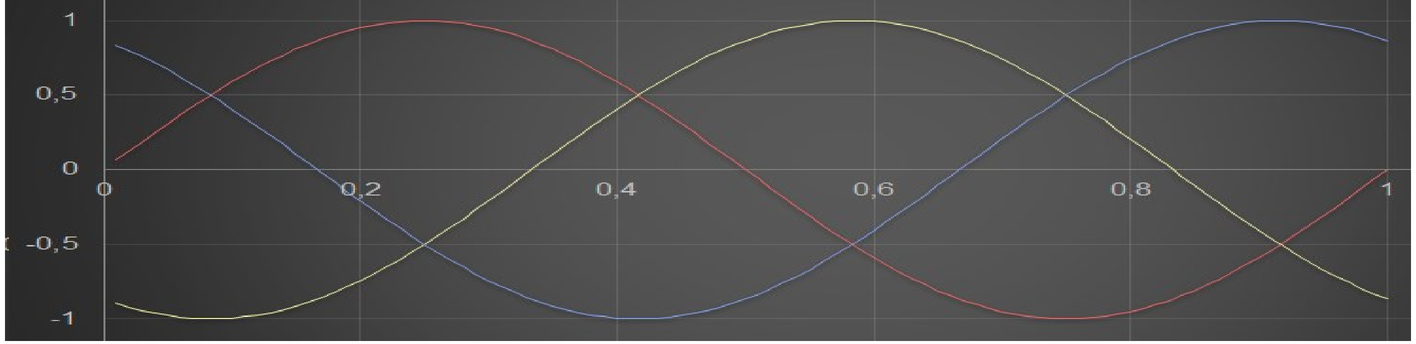


Part 3: Output control, how to achieve an output vector

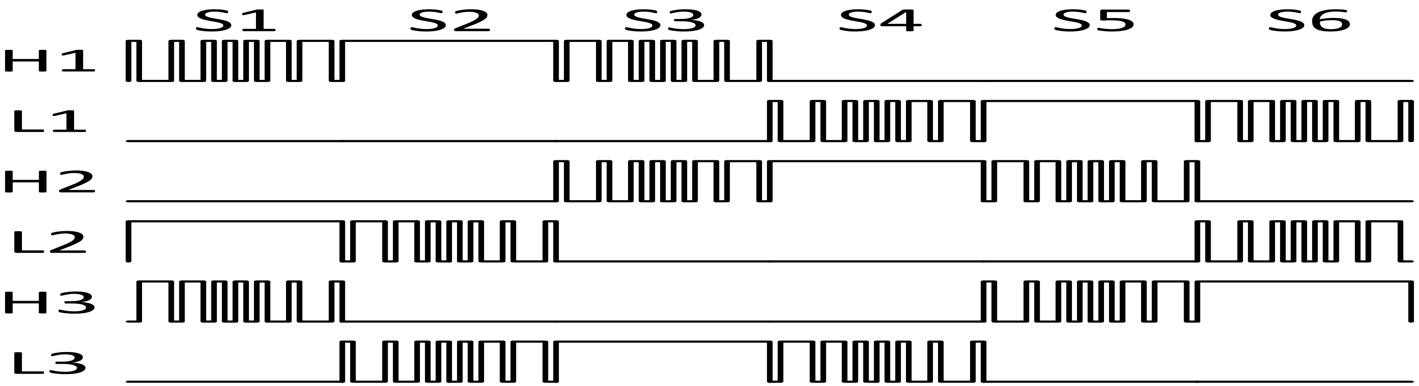
How to achieve an output vector

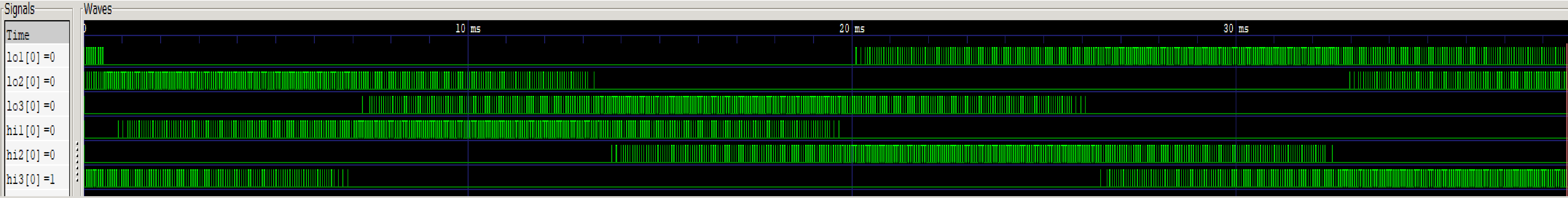
- Pulse width modulation techniques
  - Sinusoidal PWM (*S*PWM)
  - Space vector modulation (*SV*PWM)
- Motor currents in *S*PWM and *SV*PWM

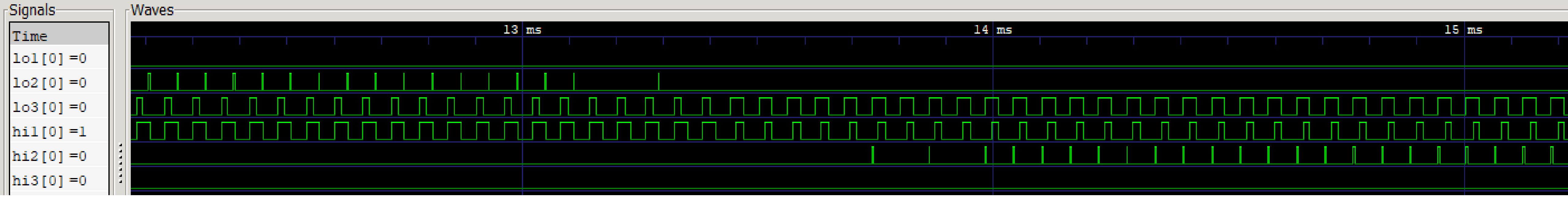
Pulse Width Modulation



- Desired output
- PWM output
  - Idea
  - Sinus modulated (SPWM)







Here: Max duty cycle at 30%

Sinusoidal PWM «SPWM»

Carrier based modulation

Power electronics handbook 4th edition, Muhammad H. Rashid 2018
ISBN : 0-12-811408-8, Butterworth-Heinemann

(Online available through Oria)

- Most PWM schemes are carrier based
  - Suited for analog comparators
  - Digitally we never create a carrier signal, but calculate PWM duty cycle directly (equivalent).

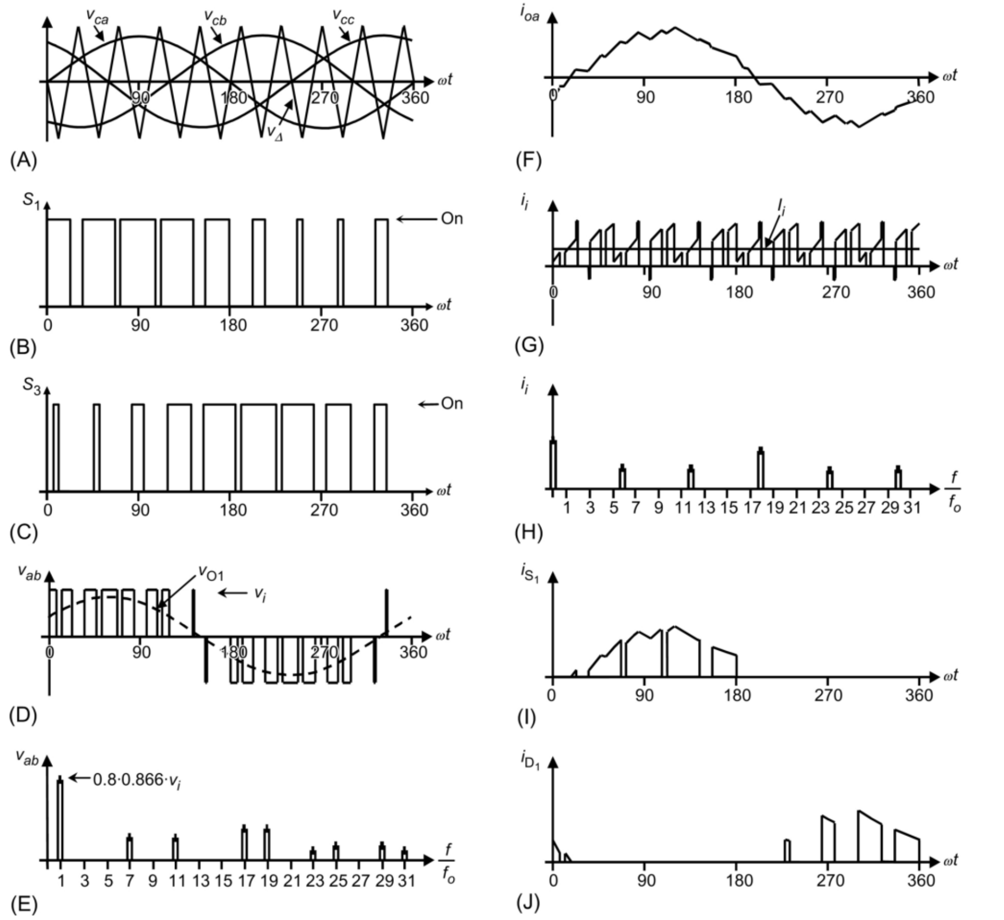
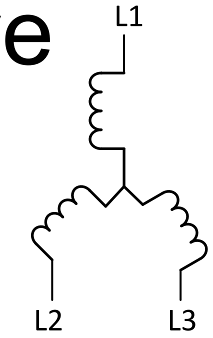


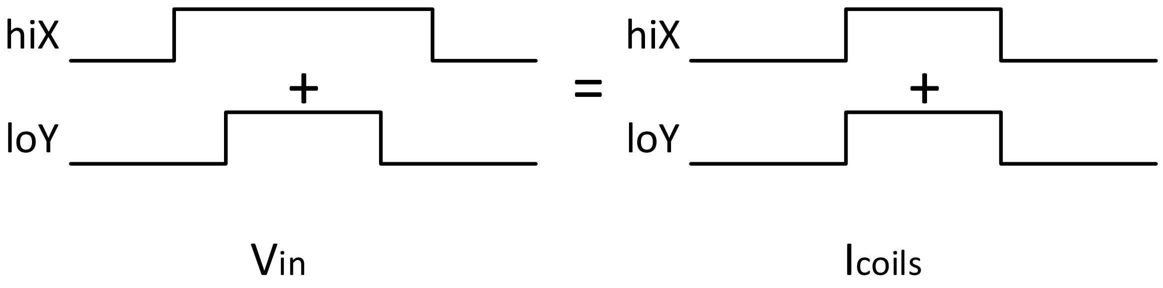
Fig. 11.18 The three-phase VSI. Ideal waveforms for the SPWM (si13_e, si14_e): (A) carrier and modulating signals, (B) switch S1 state, (C) switch S3 state, (D) ac output voltage, (E) ac output voltage spectrum, (F) ac output current, (G) dc current, (H) dc current spectrum, (I) switch S1 current, and (J) diode D1 current.

Pulse organization, Wye

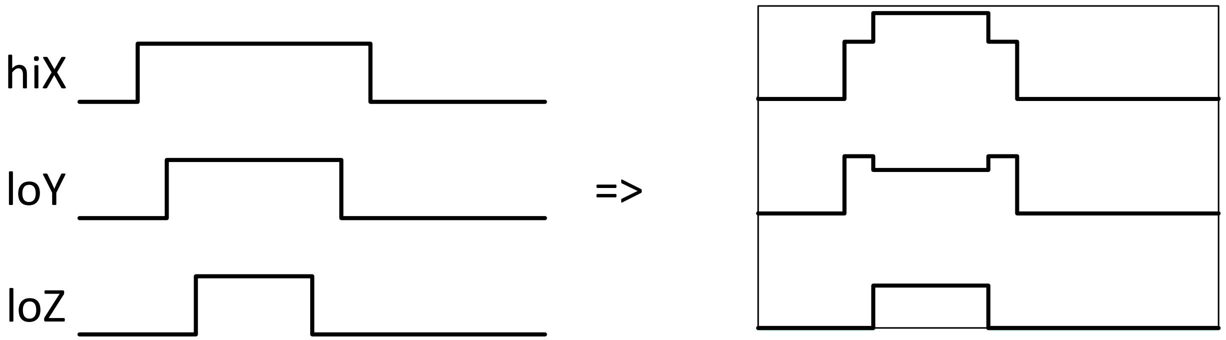


Note that

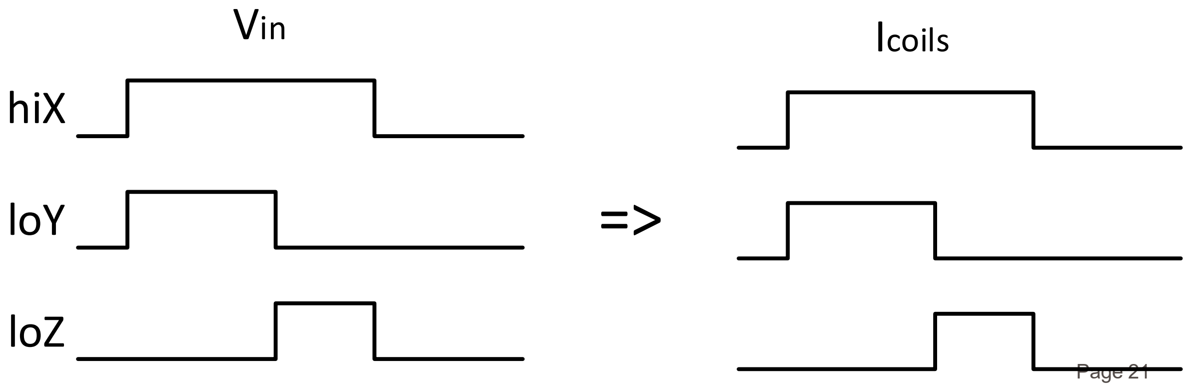
- You can't have current through one coil alone.
  - Thus…



- Directing current through three coils..
  - Does not double the total current



- Would it be an idea…
  - to put one before the other
    - *We can do this now…*
    - *No-one does this (..?)*

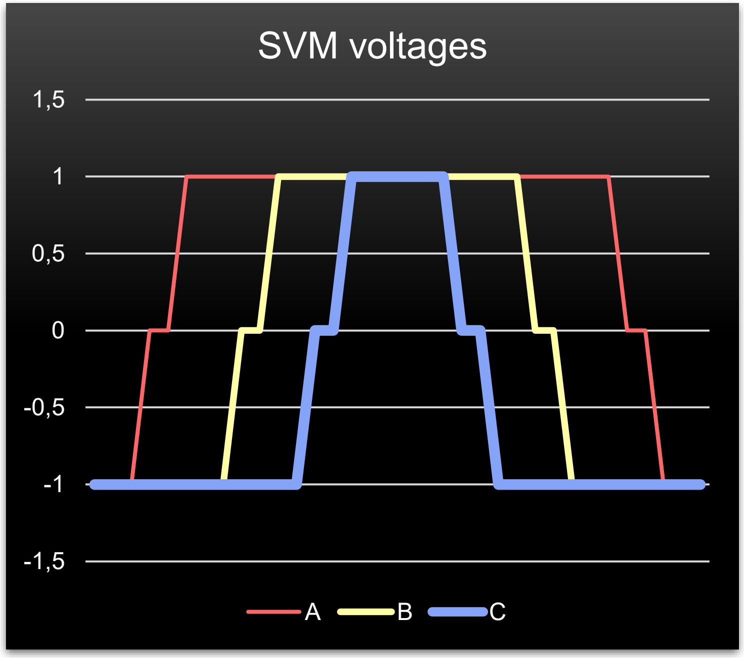


Space vector modulation -SVPWM (SVM)

The most used vector modulation

- Sinusoidal PWM:
  - Either high or low for each phase.
  - Use all 12 commutation states (and Hi-Z for 0)

- SVPWM:
  - Full voltage swing for all phases
    - Uses only 6 commutation states, never the weaker ones
    - all high or all low for 0 output.



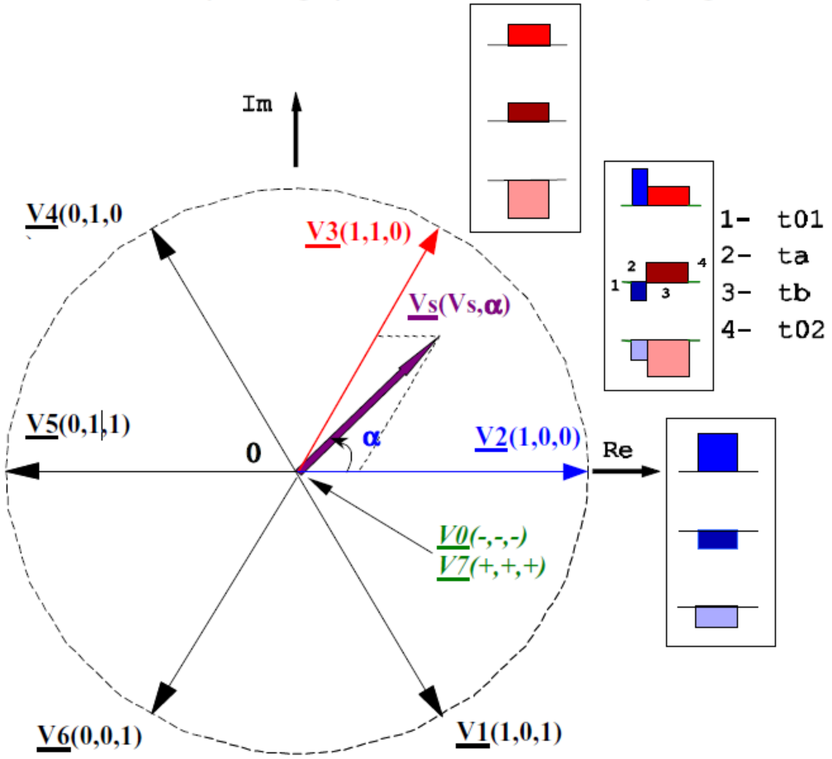
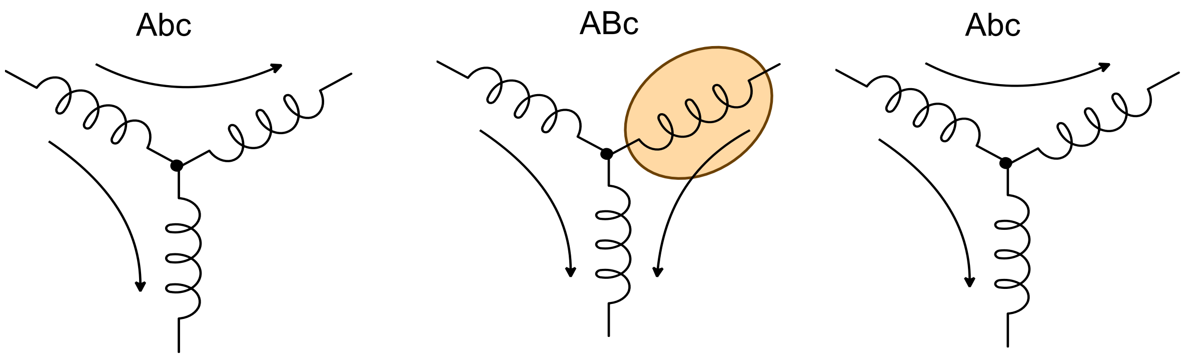


Fig.10 Generation of voltage Space Vector by SVM

SPACE VECTOR MODULATION –An Introduction

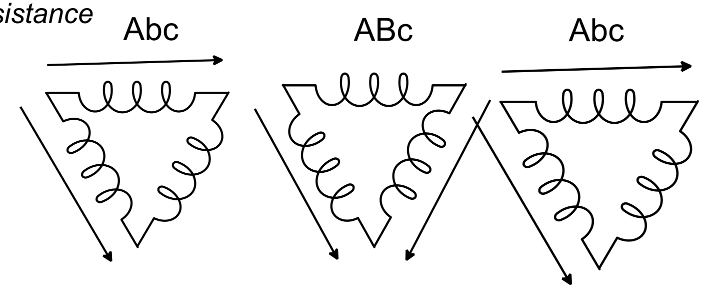
== *Tutorial at IECON2001*== Dorin O. Neacsu

SVPWM voltages and current

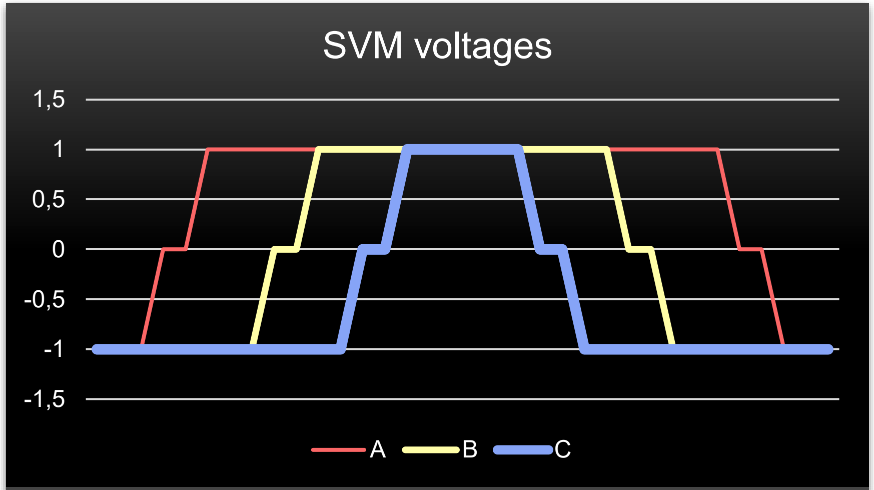


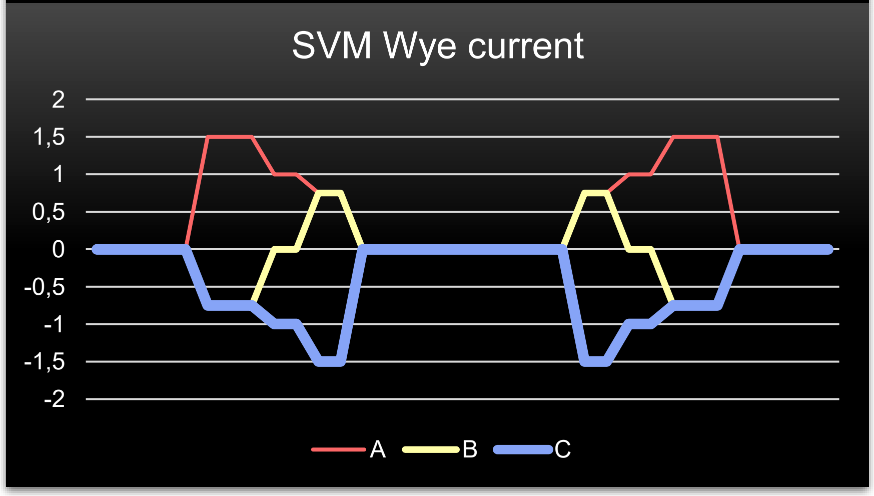
Is it really the way to go?

- All phases @ same voltage => No current
- Wye: Coil B pulls in two directions
  - All within the same PWM pulse
  - *How can this make sense?*
    - *Neglected :*
      - *coil resistance*
        - *Ideal solenoids doesnt have resistance*
        - *(it is low but not 0)*
      - *magnets being heated*
- *Delta: Not an issue*



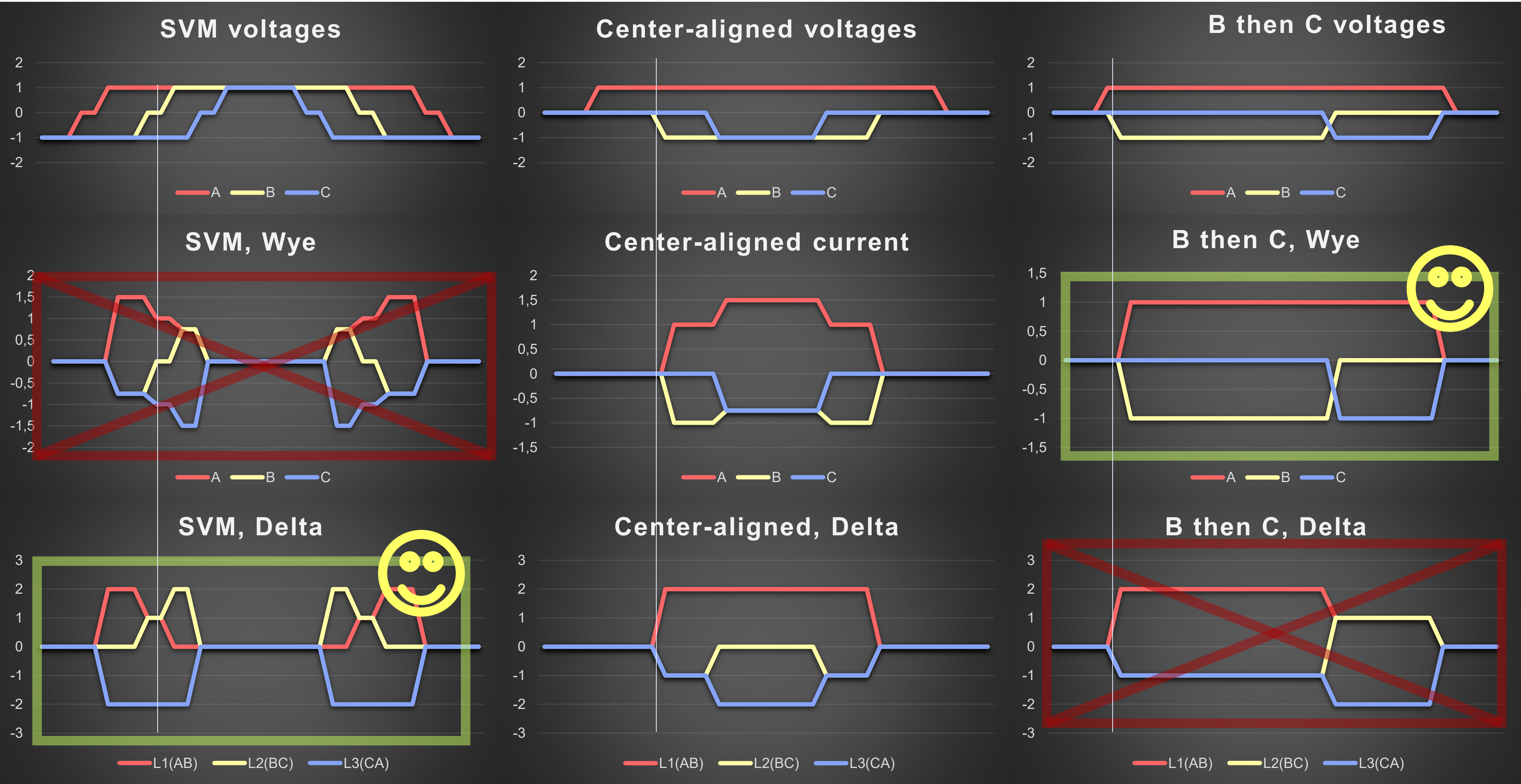
- All signals are mirrored within one period
  - The same current should is achieveable using switching only one phase (B) from high to low.
    - This constraints timing to avoid short circuit





- *Note: The exact view here is not the combination of three sinus values.*

Currents in SVM and SPWM schemes, ignoring inductance (L)



Modulations summary

Summary from previous page

- *SV*PWM goes rail to rail , but has to pause in the middle
- *SV*PWM on a Wye connection cause current turning within PWM cycle
  - => added motor resistance loss
  - => added heat in the permanent magnets
    - Often neglected in calculations
- Center-aligned *S*PWM requires more complex calculations since time is spent at ½ or 2/3 current (The other schemes can be made more powerful).
- «B then C» *not suited(?)* for Delta connection.

- The impact on efficiency of using different modulation schemes for different motors could be tested physically

Motor Control algorithm status of today..?

Motor Control algorithm status of today..?

Pulse modulation	Carrier generation
	SVPWM | SPWM | *Variants* | PDM?
	third harmonic injection
Motor configuration	Wye | Delta | Stepper motors ?
«Sensorless» sensing	High or low side phase shunt | High or low side common shunt | phase current shunt | phase voltage (EMF)
Position sensors	motor hall sensors | rotational encoders
observer circuitry	observer calculates status depending on sensors, hysteresis considerations etc
user communication	output: position (angle), speed, currents / voltages
	translate to control related values (Force torque/ etc)
	User input and (PID) control
	translate force or speed to current (assumed tangential)
	input: desired Torque/force (always assumed tangential)
Low level motorControl	Clark-Park transformation (Direct-quadrature-zero transformation, DQZ):
	convert three phase EMF-current to XY plane
	convert EMF-current to radial/ tangential
	PI control of tangential (∝ F) and radial component (set to 0)
	convert radial/tangential currents to XY plane
	convert XY plane currents to three phase
	Convert currents to pulse length = Pulse modulation

Motor Control algorithm status of today..?

Pulse modulation	Carrier generation; SVPWM / SPWM / *Variants* / PDM?; third harmonic injection	Low frequencies better for driver, high frequencies better for motor. Will affect measurment options	How can we determine optimal frequency or pulse lengths? IS PDM possible to achieve while maintaining good measurments and regularity?
Motor configuration	Wye / Delta / Stepper motors ?	May require entirely different pulse calculation	Will a stepper motor configuration have less loss due to individual coil actuation?
«Sensorless» sensing	High or low side phase shunt / High or low side common shunt / phase current shunt / phase voltage (EMF)	Depends on power driver circuit configuration. Timing must be coordinated with PWM	Which measurement schemes will allow us to have the best possible control system? (will affect PCB layout)
Position sensors	motor hall sensors / rotational encoders	Hall sensors and EMF may have hysteresis (direction, speed, pulses?)	Do we need to calculate hysteresis for control purposes when using encoder data?
observer circuitry	observer calculates status depending on sensors, hysteresis considerations etc		Should hysteresis be learnt once, or continously?
user communication	output: position (angle), speed, currents / voltages; translate to control related values (Force torque/ etc); User input and (PID) control; translate force or speed to current (assumed tangential); input: desired Torque/force (always assumed tangential)	Matrix multiplication (float multiplication)	Is ML desireable for calculating hysteresis; Can we use ML to learn motor wear?
Low level motorControl	Clark-Park transformation (Direct-quadrature-zero transformation, DQZ): convert three phase EMF-current to XY plane; convert EMF-current to radial/ tangential; PI control of tangential (∝ F) and radial component (set to 0); convert radial/tangential currents to XY plane; convert XY plane currents to three phase	Matrix multiplication (incl 4 sinus calc); PI calculation (multiplication / addition); Matrix multiplication (incl 6 sinus calculations, float addition and float multiplication)	Should we do without clark-park using digital angles? (0-360 = 0-xFF); Can we do with a single PID controller?
	Convert currents to pulse length = Pulse modulation		

Thanks for watching!