How to Design, Simulate, and Manufacture a Self-Balancing Robot

David Truyens

Application Engineer

Twitter: @davidtruyens

Key learning objectives

At the end of this class, you will be able to:

- Learn how to simulate a self-balancing robot within Inventor DS
- Learn how to use this technology to solve real-world challenges
- Discover what the speaker has tried and what didn't work
- Discover the future of making machines

Why?

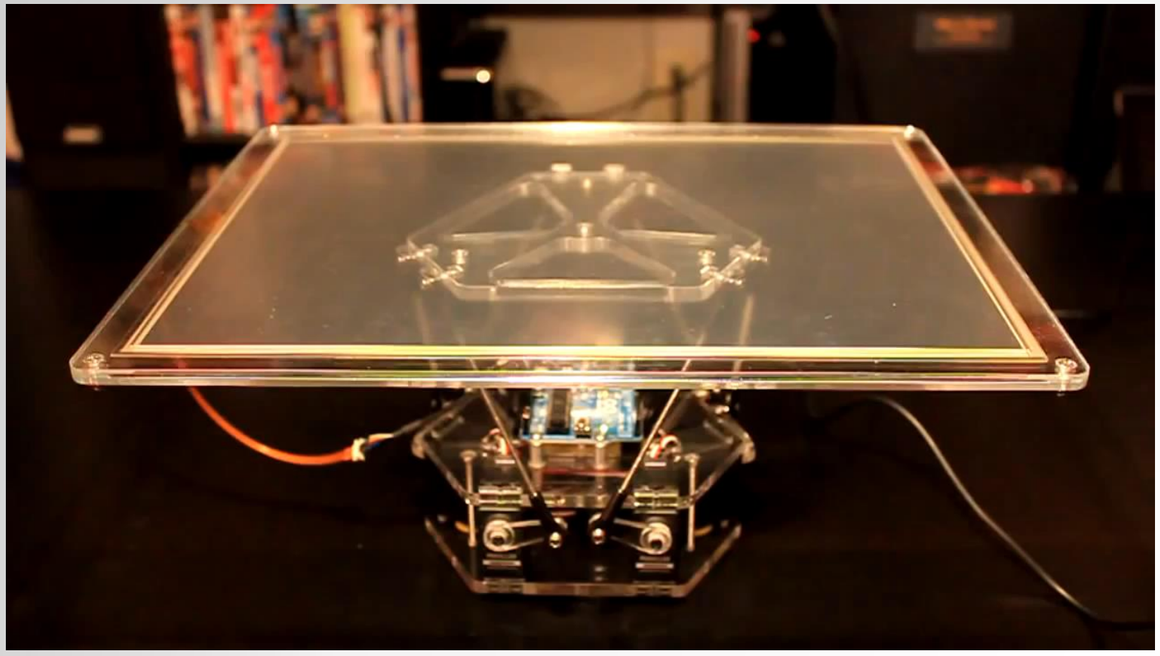
Why you shouldn’t …

- Uncountable hours
- Money
- Less sleep
- Headache
- Relationship

Why you should!

- Controllers are everywhere!
- Everything is possible these days
- FOMT

Gadgets



Full Motion Dynamics

Flanders Make

Balanduino / TKJ Electronics

Arduino - Massimo Banzi

Real stuff

Ampelmann

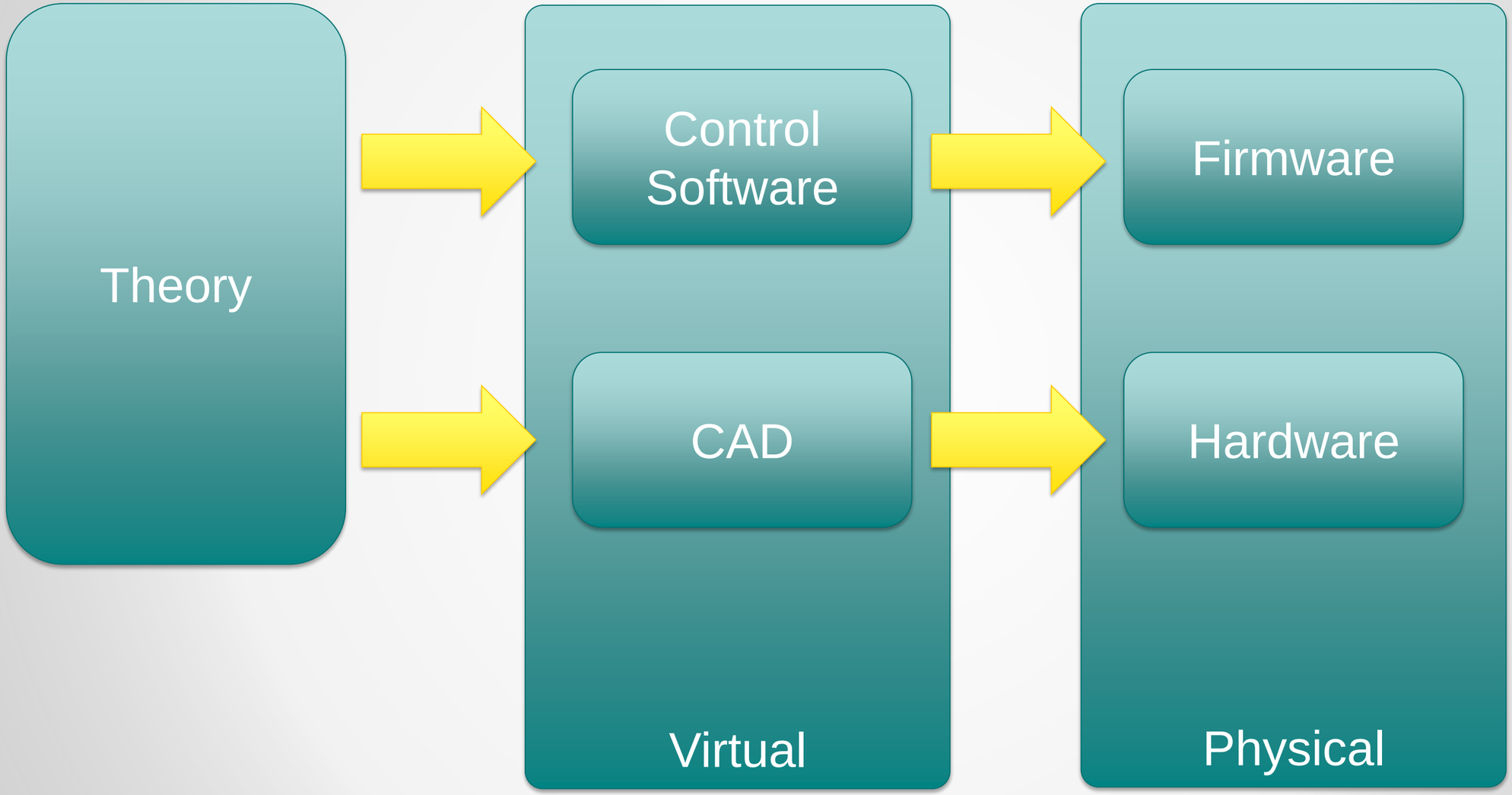
hibbot

Hibbot

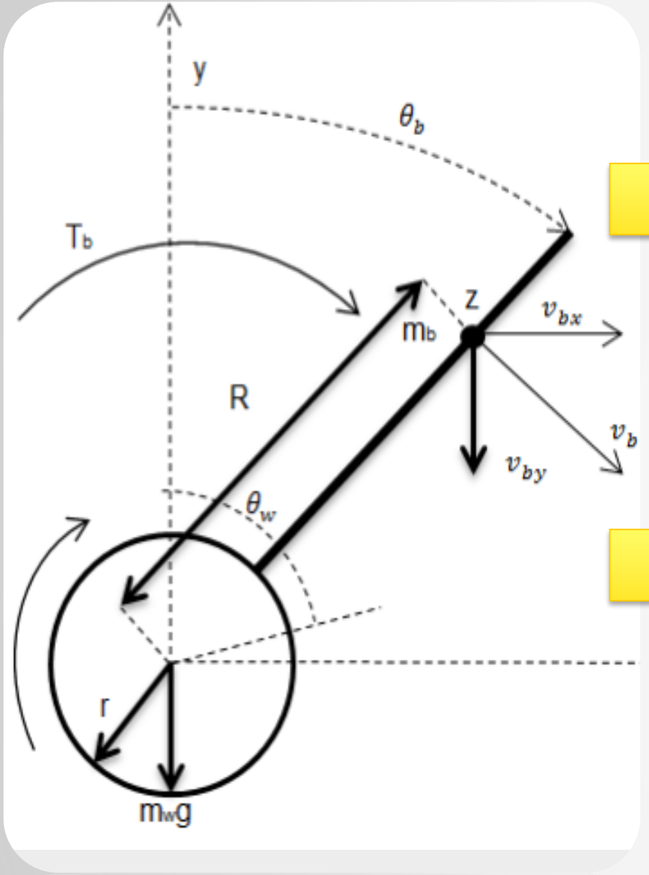
Space and automotive

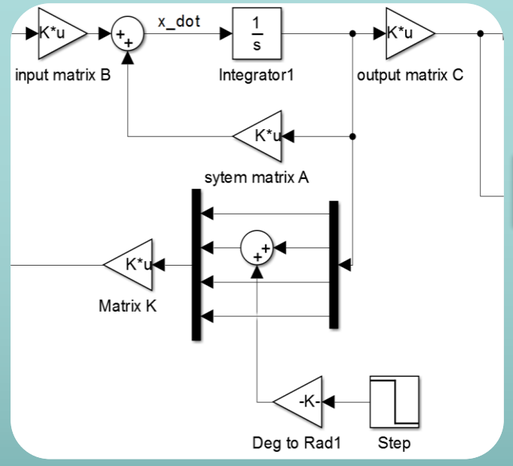
Why you should

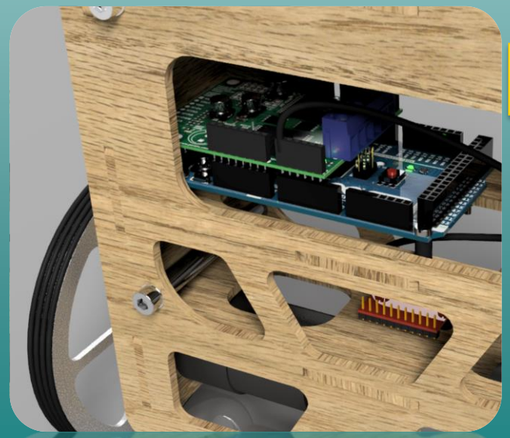
A self balancing robot



A self balancing robot

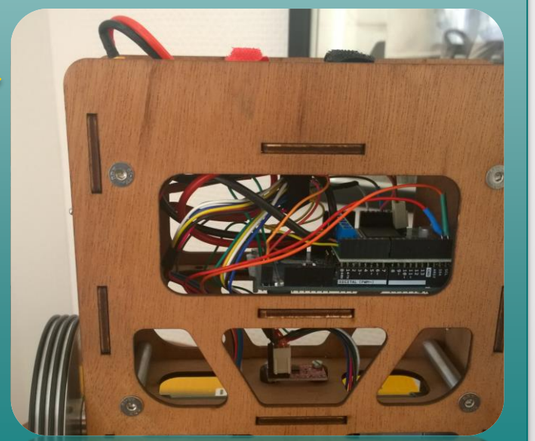






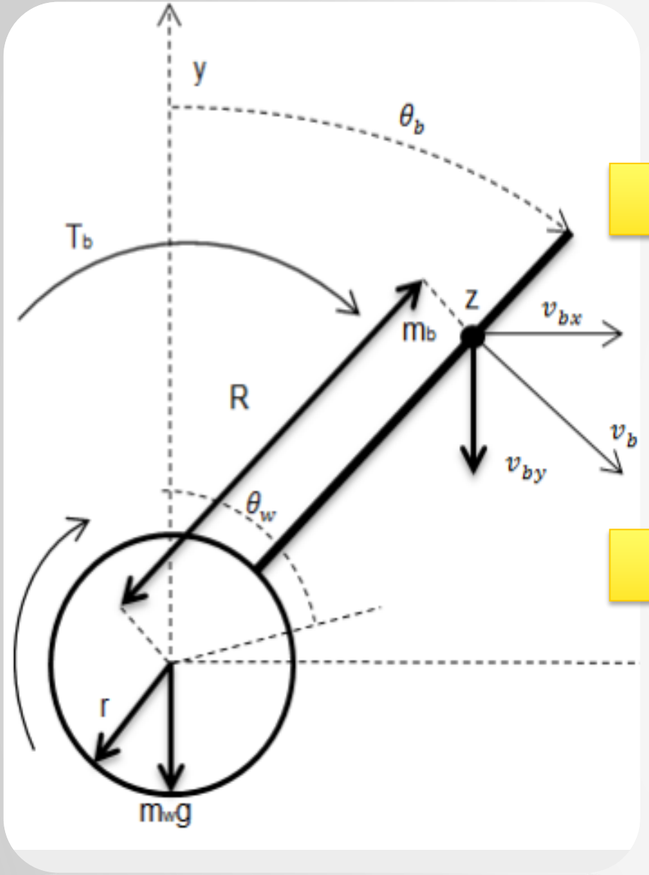
Virtual

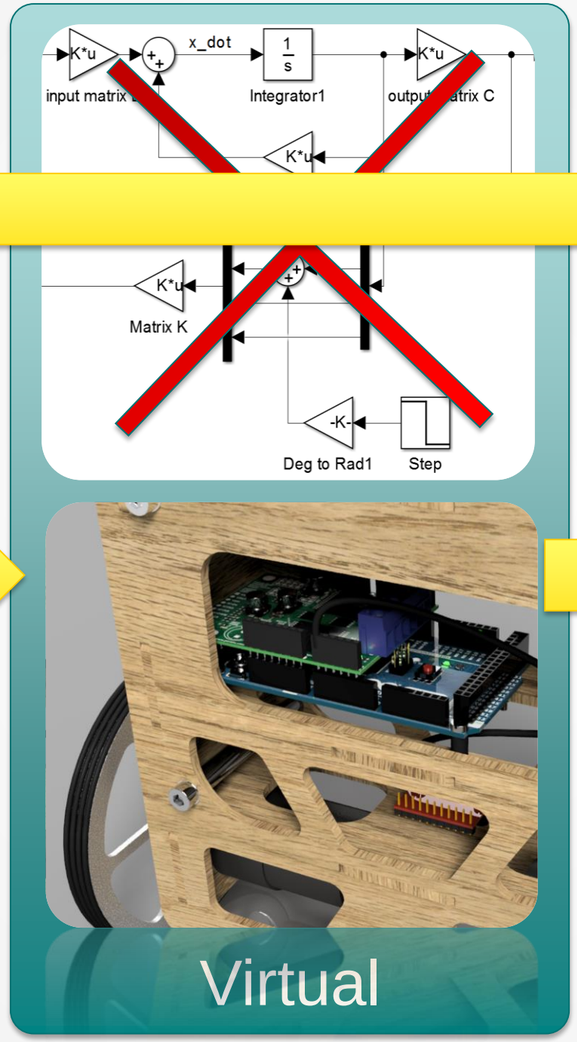
```
void calibrateMotor() {

  Serial.println(F("Put the robot so the wh
  while (Serial.read() == -1);

  Serial.println(F("Estimating minimum star
  delay(2000);

  minSpeed();

  double leftSpeed = 10, rightSpeed = 10;
  testMotorSpeed(&leftSpeed, &rightSpeed, 1

  Serial.print(F("\r\nThe speed values are
  Serial.print(leftSpeed);
  Serial.print(F(","));
  Serial.println(rightSpeed);
```

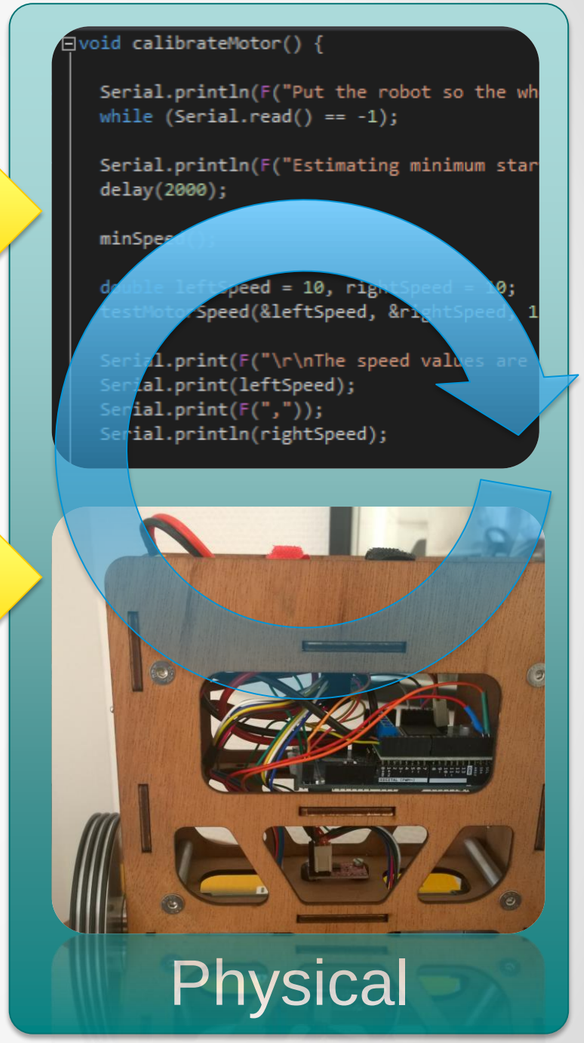


Physical

Garage version



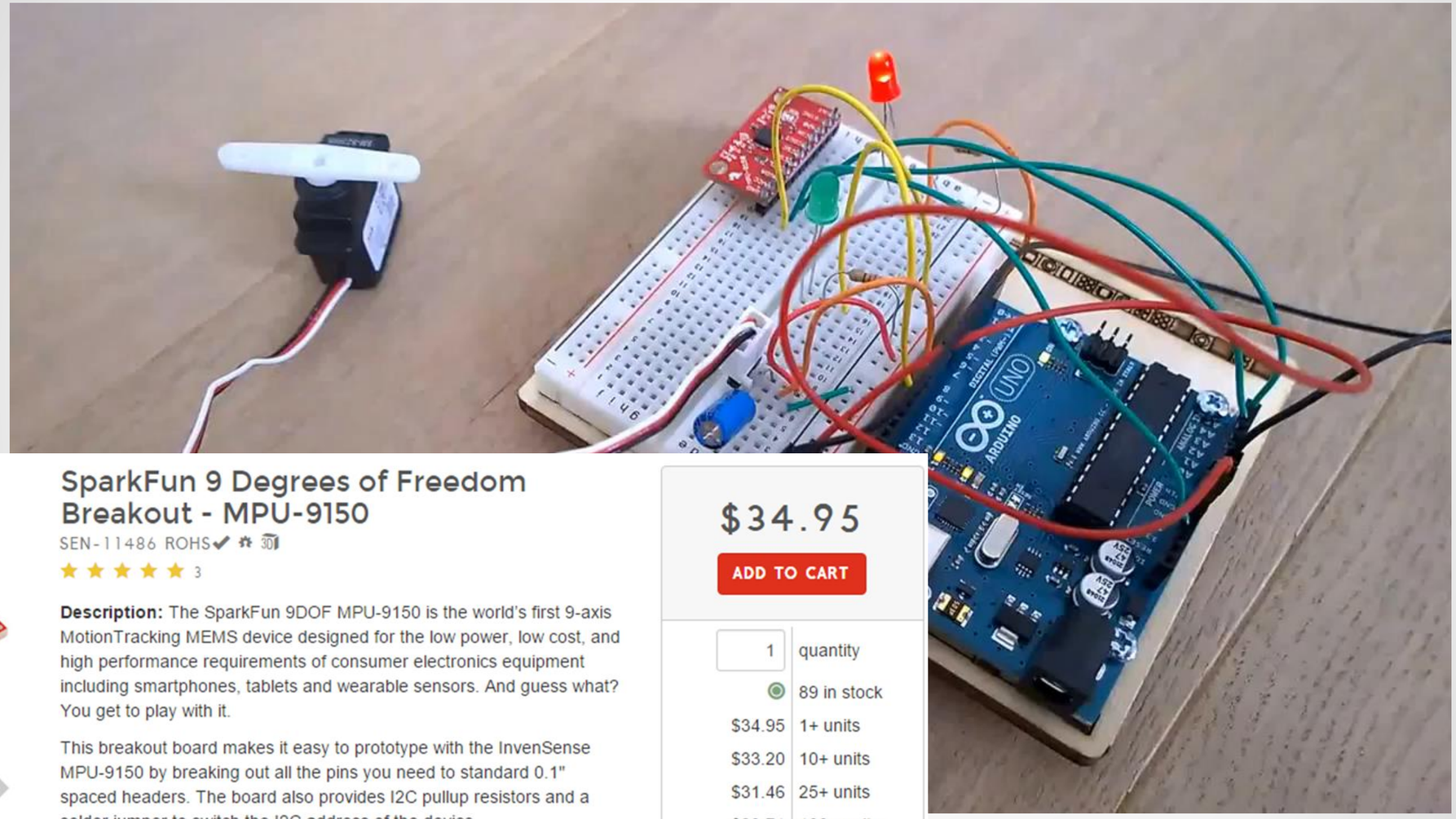


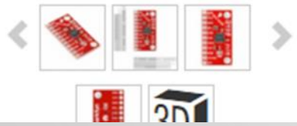


Garage Version

- Open Pandora’s box:

Garage Version





SparkFun 9 Degrees of Freedom Breakout - MPU-9150

SEN-11486 ROHS

★★★★★ 3

Description: The SparkFun 9DOF MPU-9150 is the world's first 9-axis MotionTracking MEMS device designed for the low power, low cost, and high performance requirements of consumer electronics equipment including smartphones, tablets and wearable sensors. And guess what? You get to play with it.

This breakout board makes it easy to prototype with the InvenSense MPU-9150 by breaking out all the pins you need to standard 0.1" spaced headers. The board also provides I2C pullup resistors and a solder jumper to switch the I2C address of the device.

The MPU-9150 is a System in Package (SiP) that combines two chips:

$34.95

ADD TO CART

1	quantity
	89 in stock
$34.95	1+ units
$33.20	10+ units
$31.46	25+ units
$29.71	100+ units

9 DoF MPU?

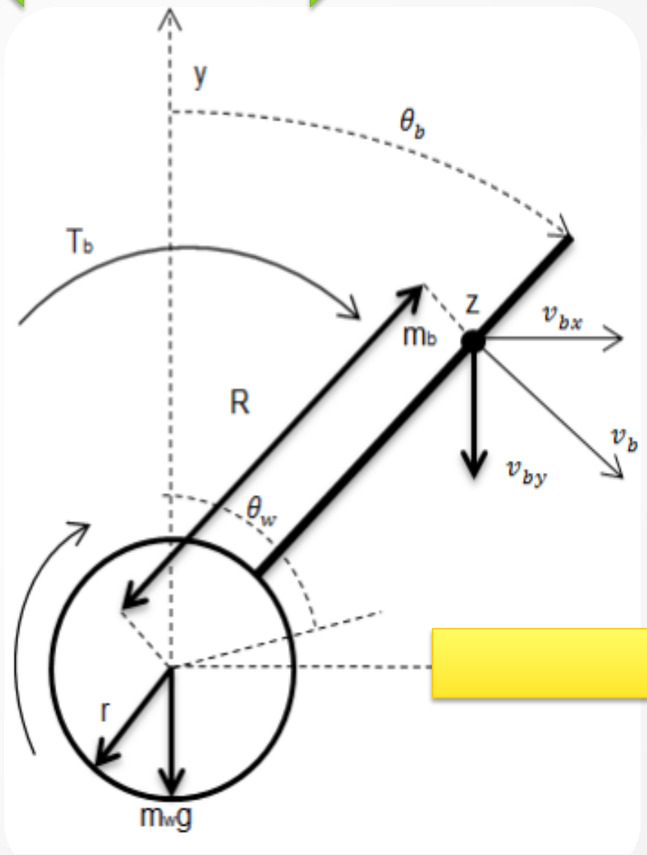
- Gyro (electronical) = angular velocity
  - Very accurate!
  - No angle!!
- Accelerometer = gravity and other accelerations
  - Direct angle!
  - Lot's of noise…
- Compass = North direction in an XYZ vector
  - Gravity independent
  - Slow and not accurate

Kalman Filter

PID Controller

- **P** proportional
- **I** integral
- **D** derivative

$* \int balance\ error$

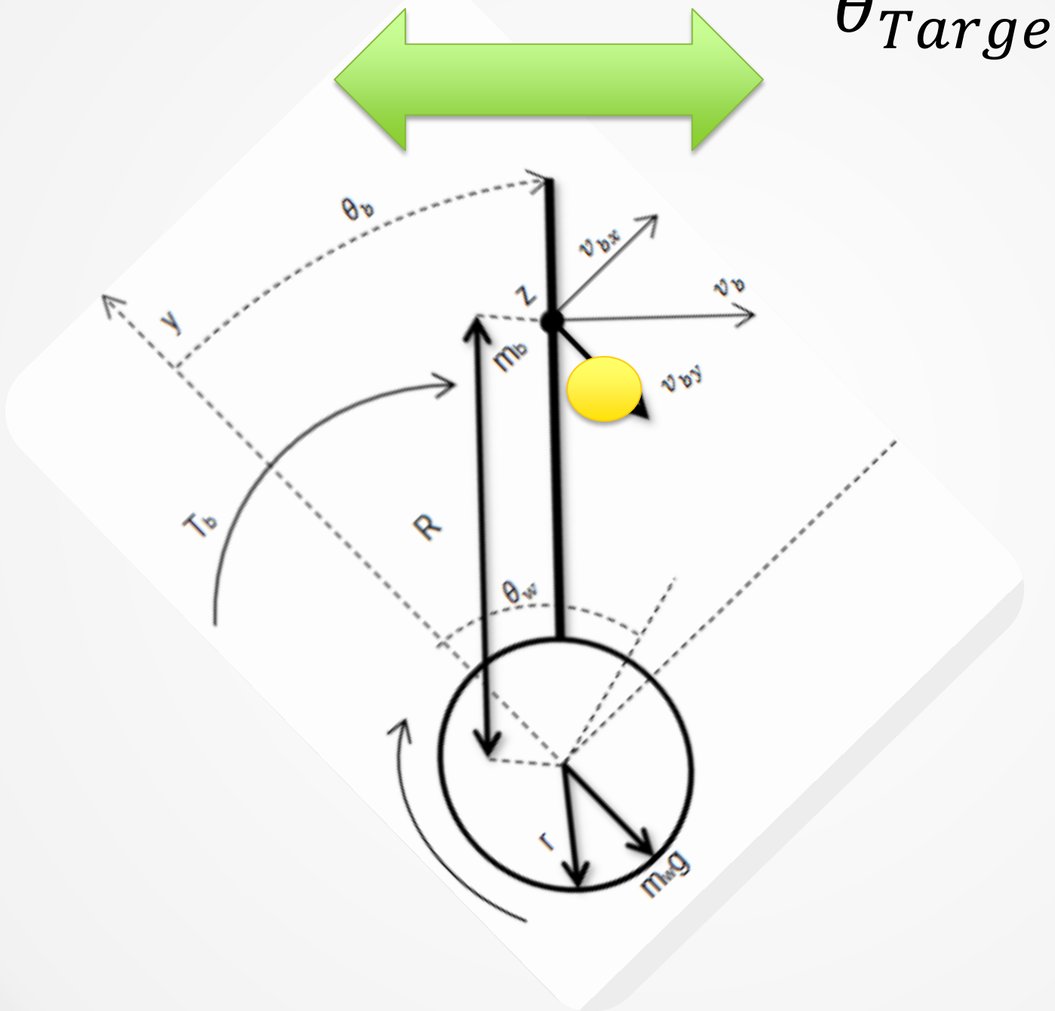


$D * \dot{\theta}_b$

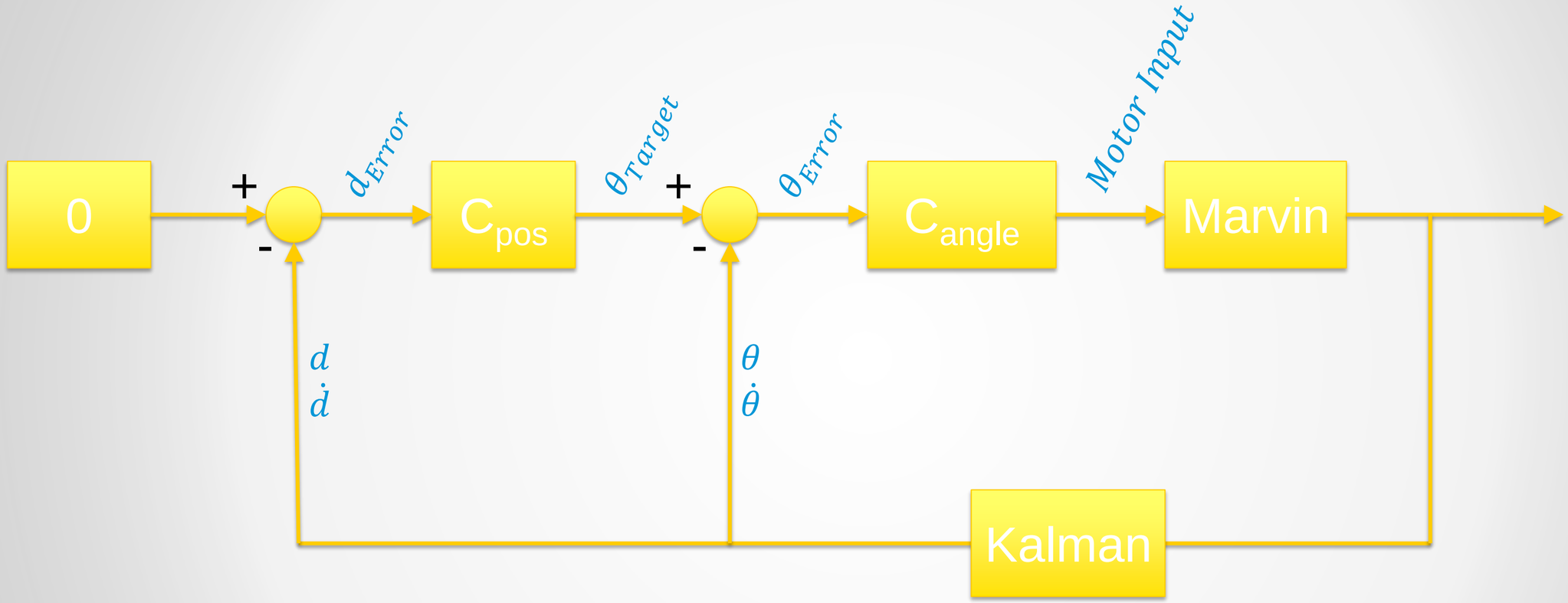
$P * \theta_b$

Master Slave - PID Controller

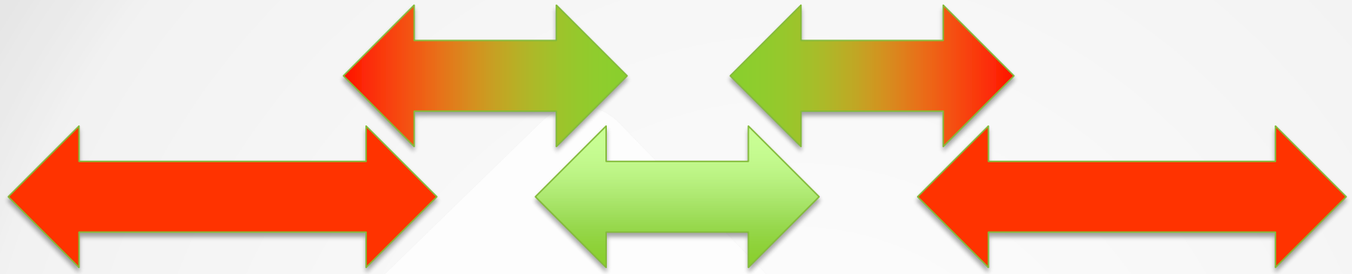
$$\theta_{Target} = P_{pos} * \Delta d \; - D_{pos} * v$$

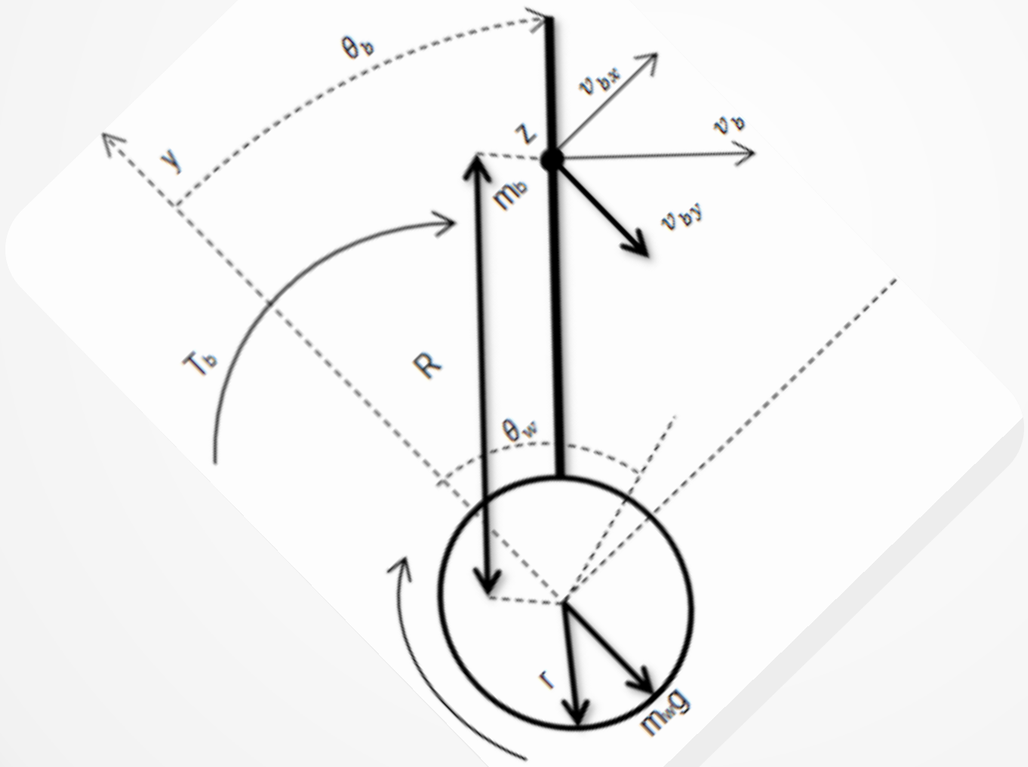


Master-slave controller

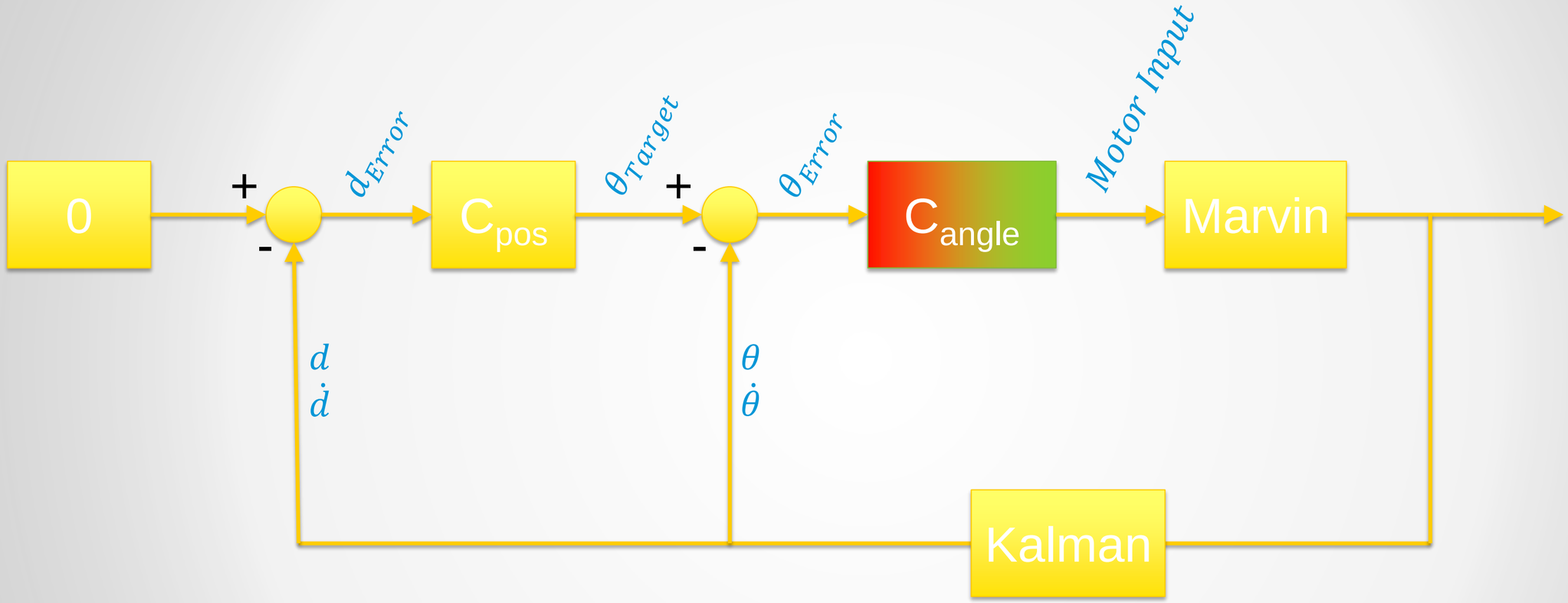


Next challenge…. Gearboxes…



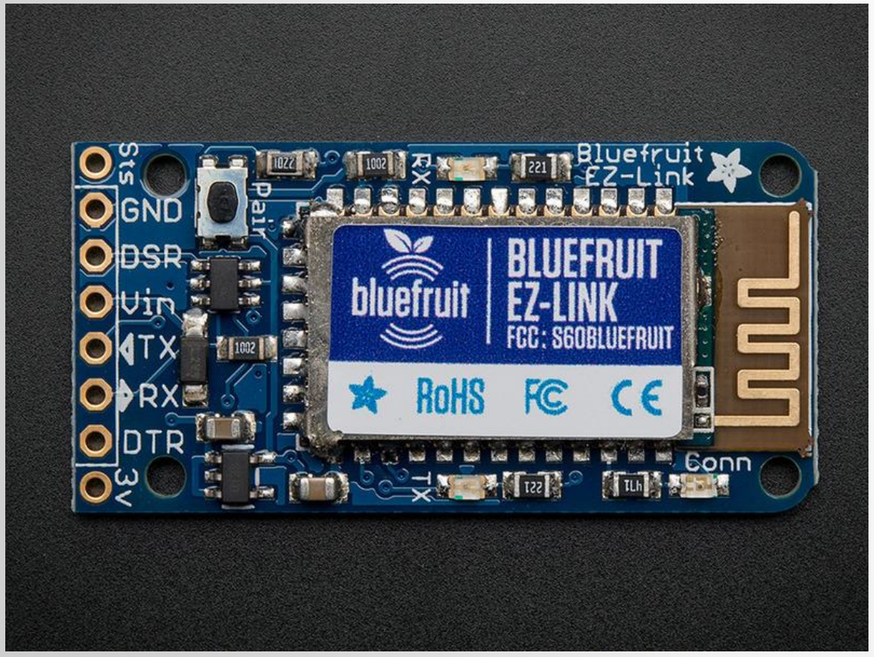


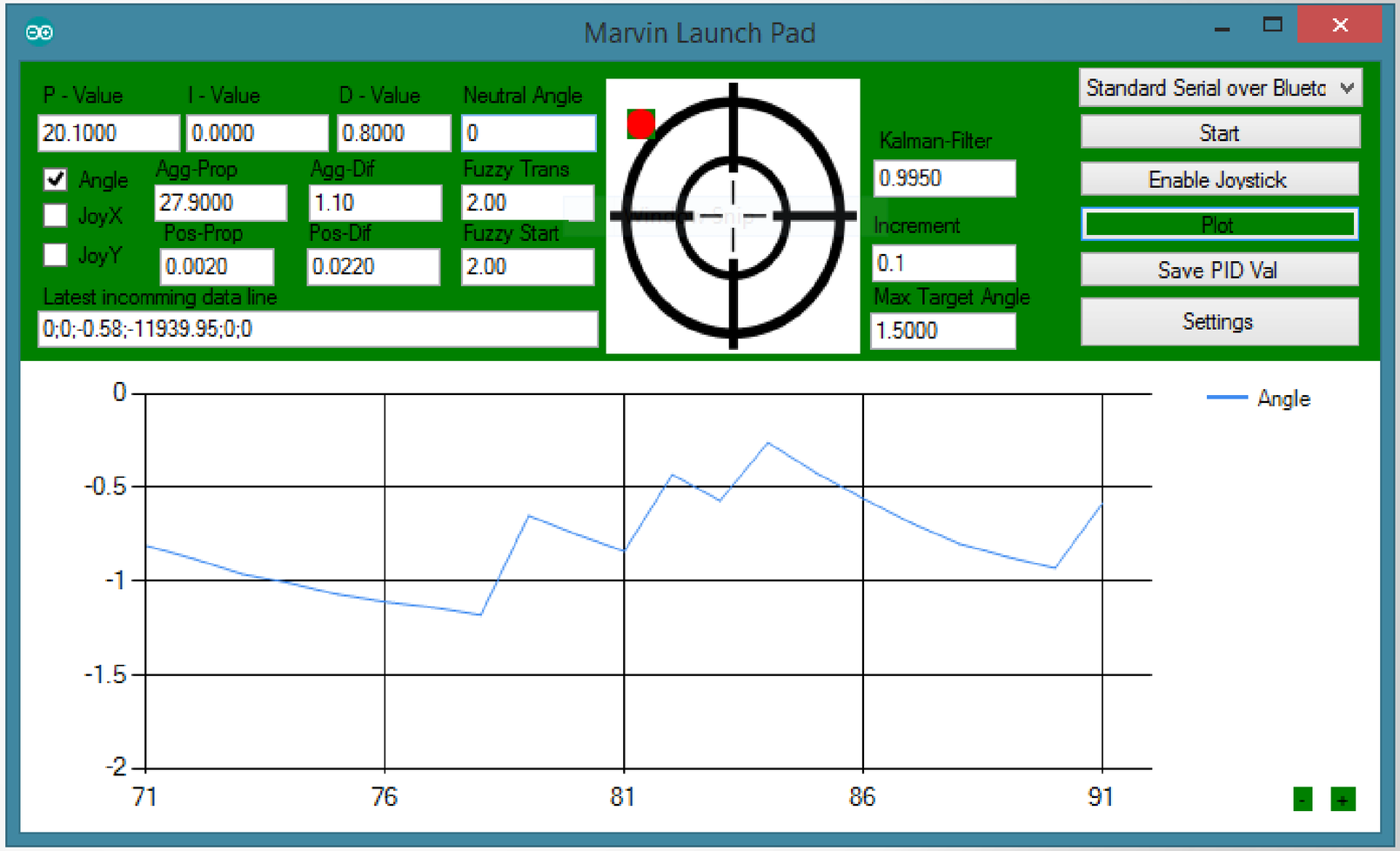
Master-slave fuzzy-logic controller



$$P_{angle}\ \ D_{angle}\ \ P_{angle}\ \ D_{angle}\ \ P_{pos}\ \ D_{pos}\ \ Kalman$$

Marvin Launch Pad





Garage Version

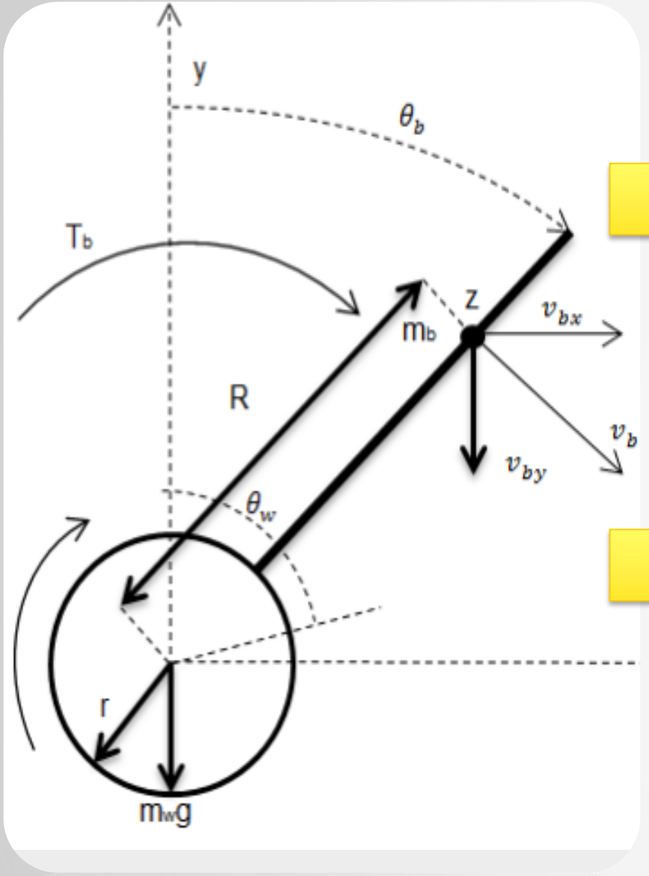
Advantages

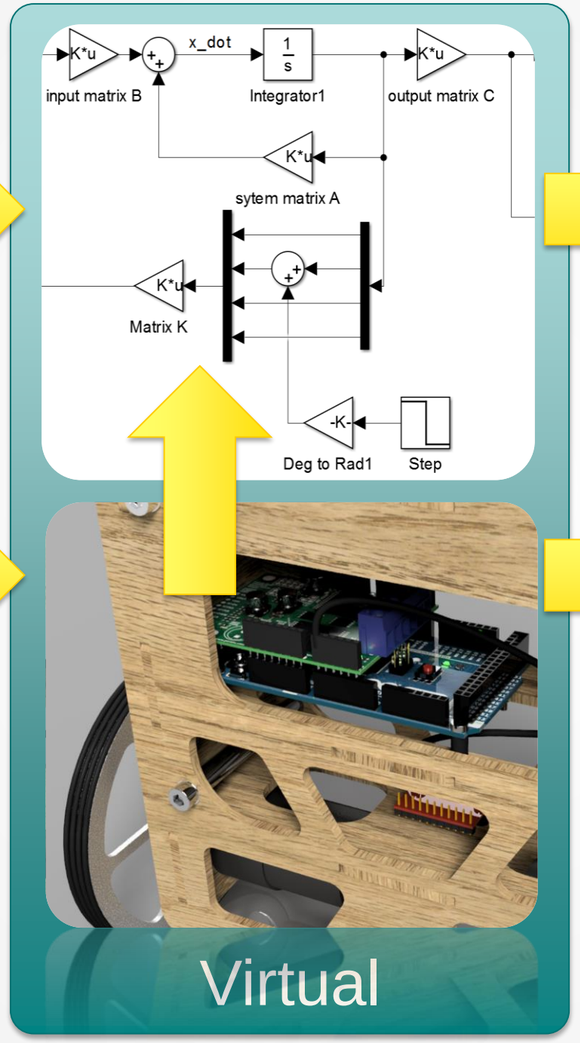
- Bottom up – step by step
- Quick results
- Not much theoretical background needed
- Fun / educational

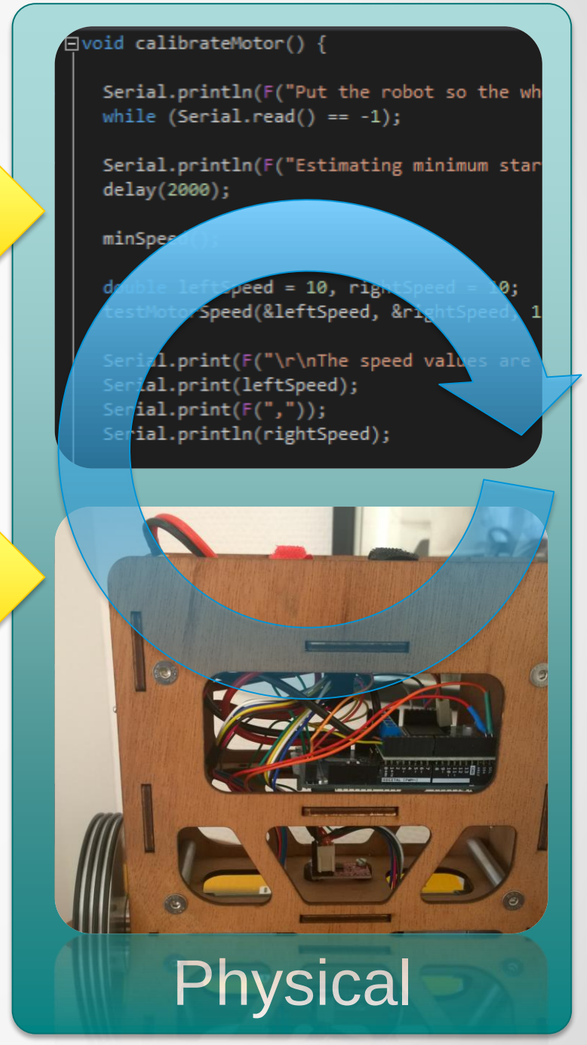
Disadvantages

- Trial and error, error, error
- Not possible for large scale projects
- Limited in complexity

Classical approach
Mathlab – Simulink - Simmechanics







Classical approach

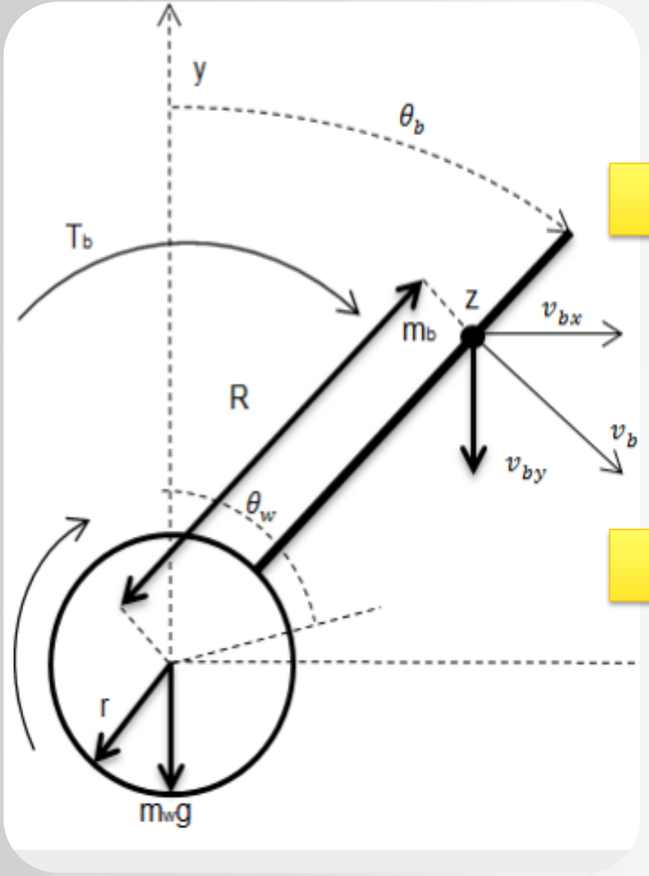
Advantages

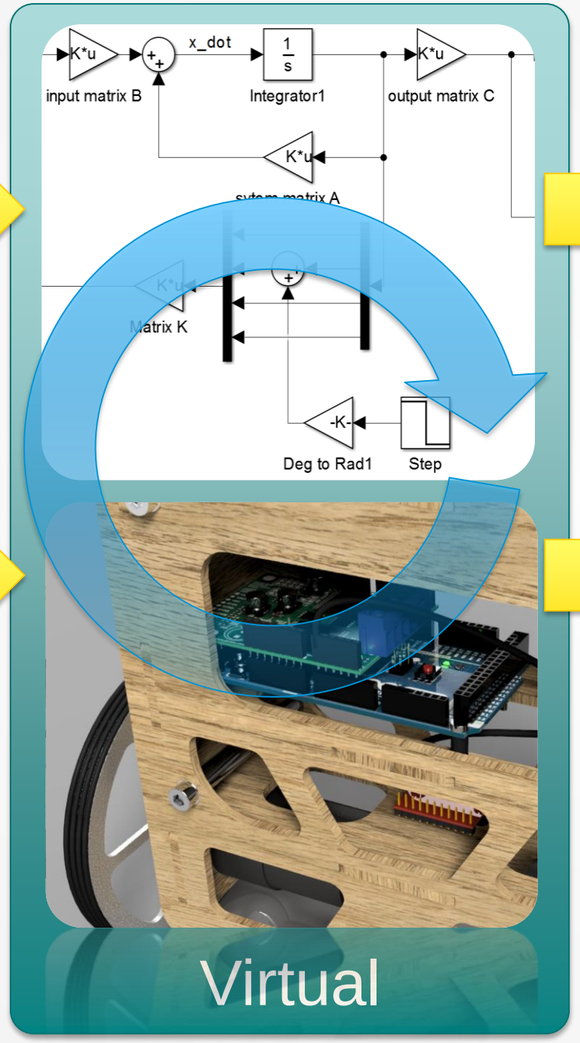
- Industry standard
- Lots of options
- Option to link cad data

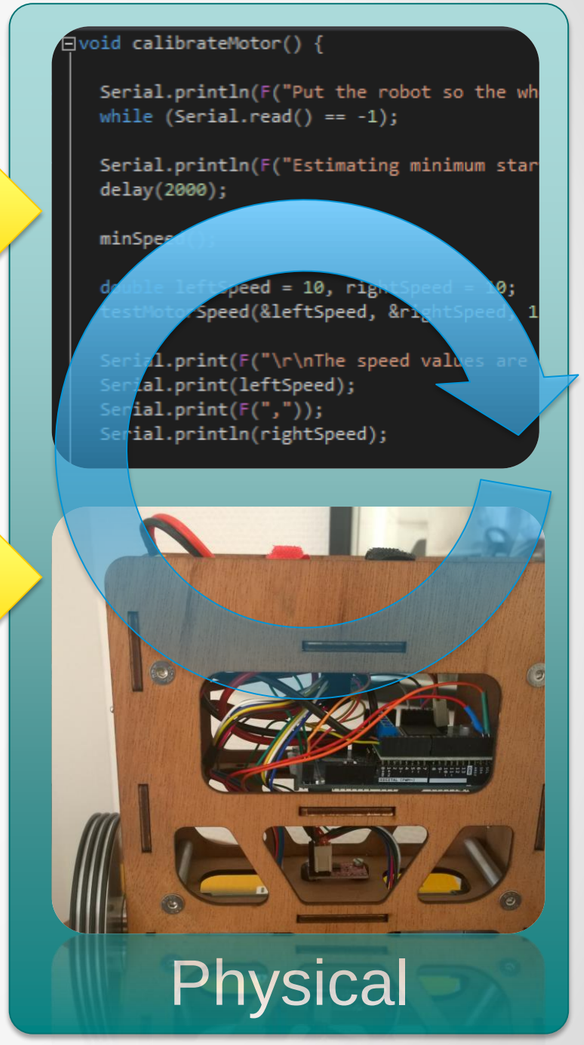
Disadvantages

- Broken workflow in Inventor
- Complicated
- Easy to make mistakes
- No validation

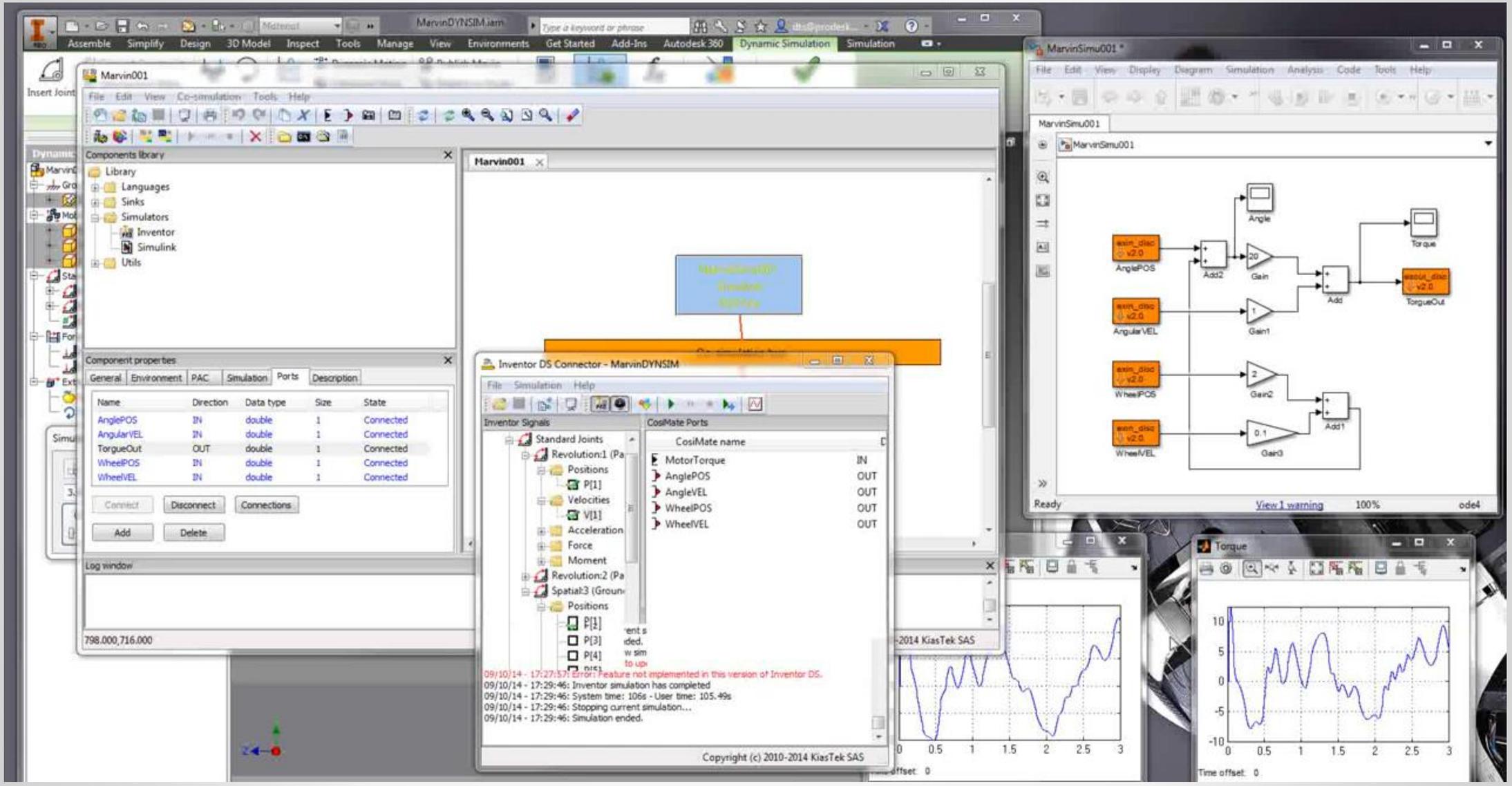
Co-Simulation approach
Mathlab – Cosimate - Inventor







Co-Simulation approach



Co-Simulation approach

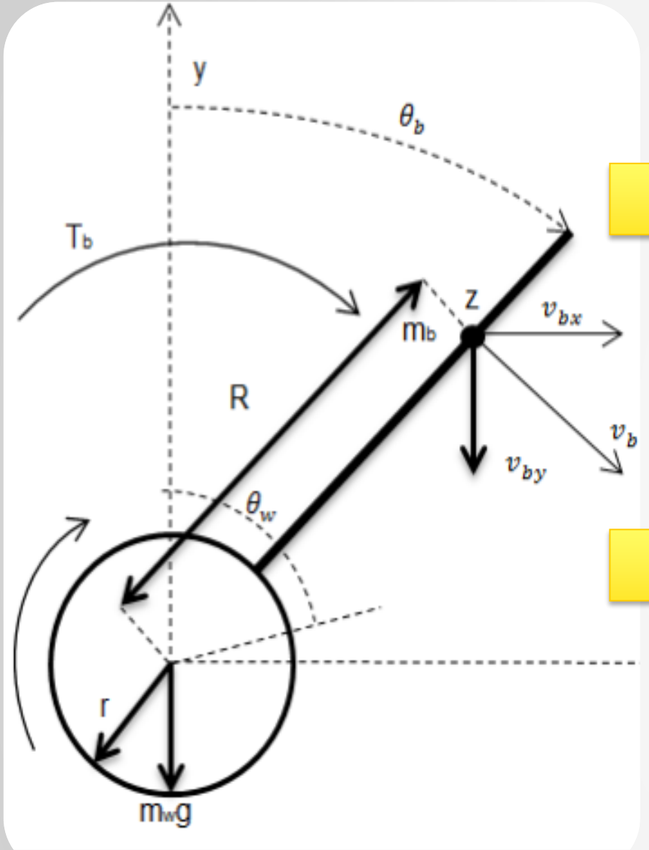
Advantages

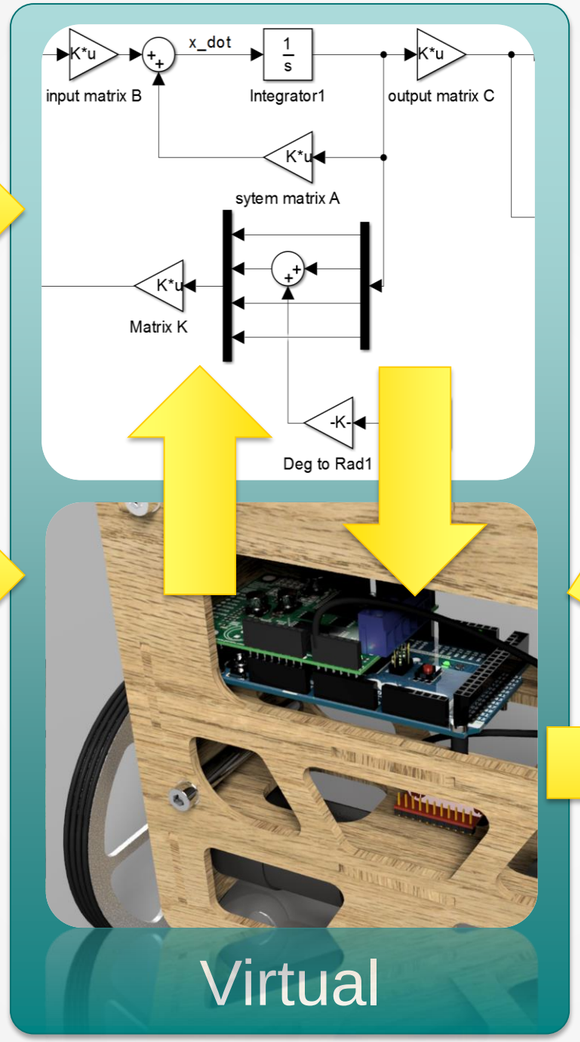
- Connected worfklows with Inventor and Mathlab
- Industry standard
- Collaborate with multiple users (Cosimate)

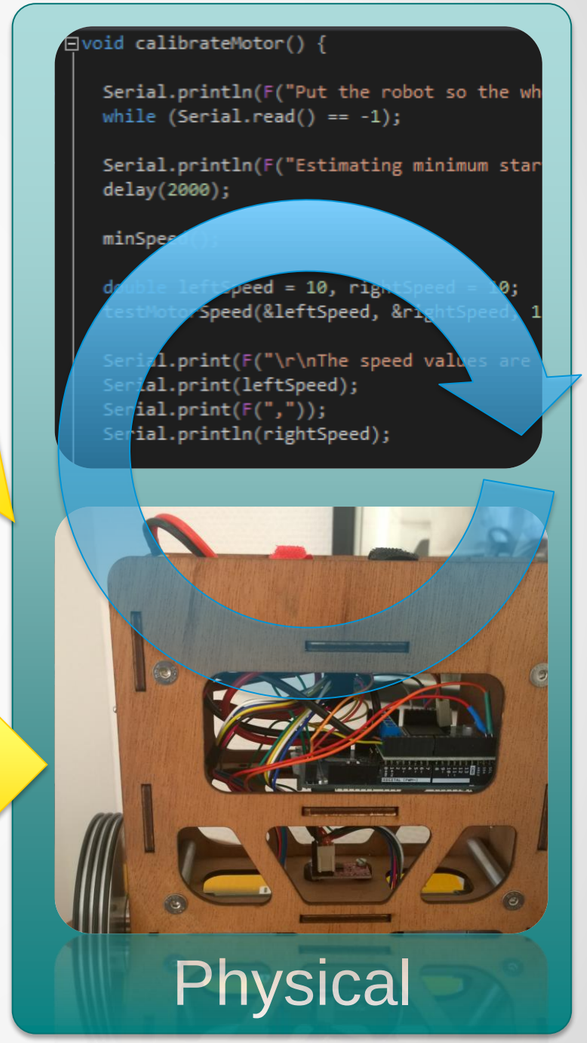
Disadvantages

- Slow
- Complicated
- Easy to make mistakes
- No validation

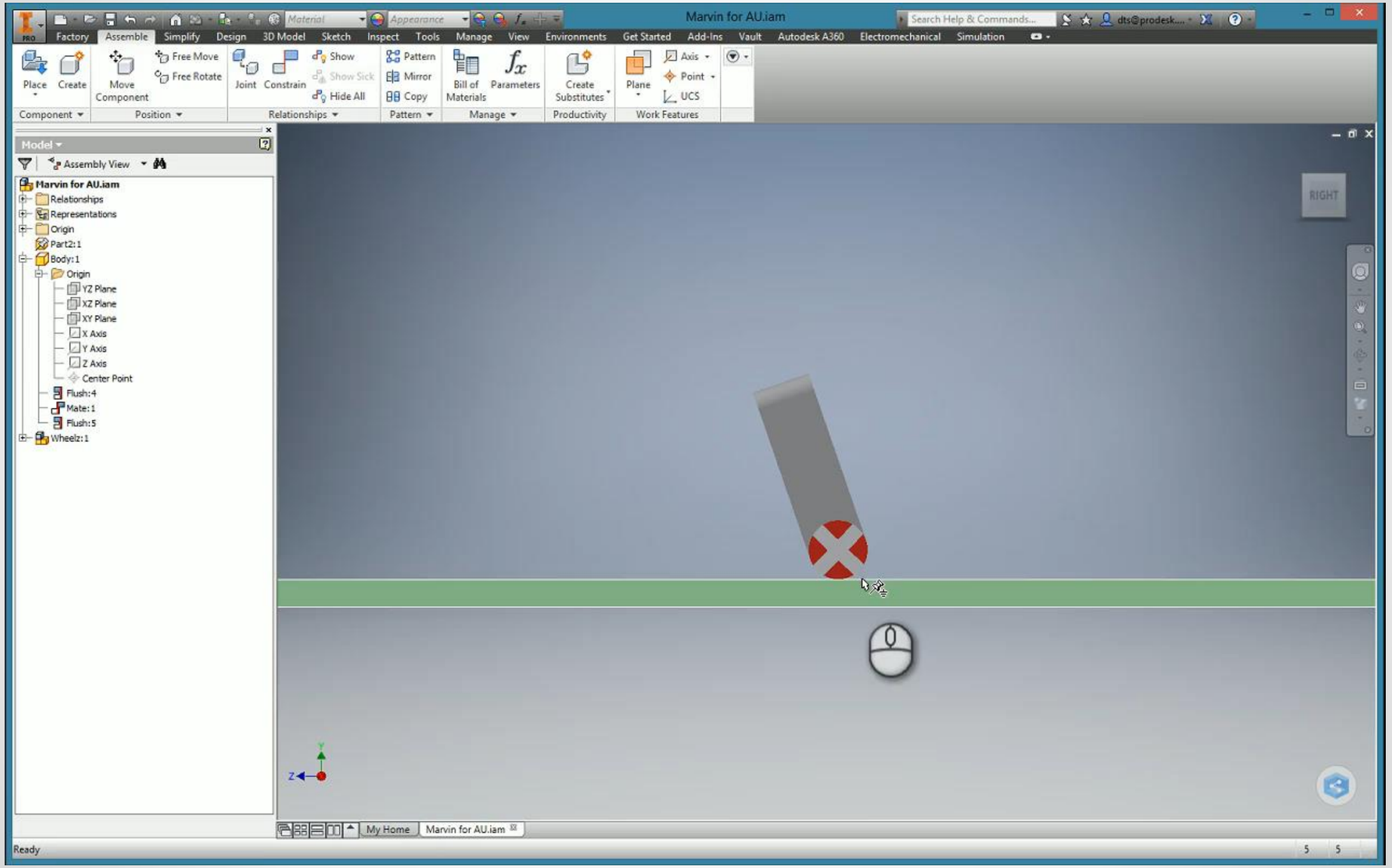
Virtual validation workflow







Virtual validation workflow



Virtual validation workflow

Advantages

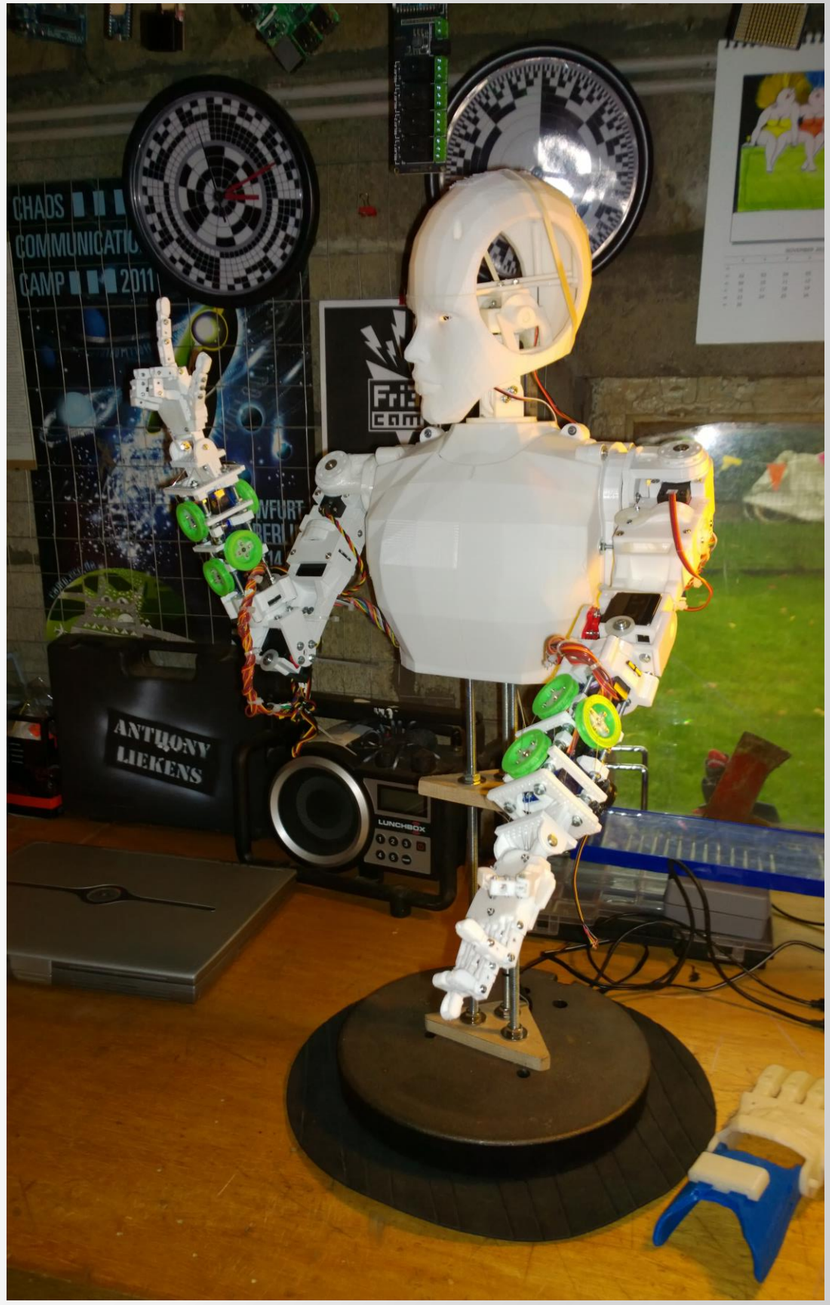
- Fully integrated in Inventor
- Validation of the control software
- Lots of potential using iLogic

Disadvantages

- Quite hidden
- Not easy to program
- Strange things when programming

Conclusions

- Lots of way’s to work
- Make your hands dirty



Future

- Collaborate with Autodesk on Smart Machines
- Create my own pcb with Circuits IO
- Make the next Marvin

Next generation

Open call

Inventor in Motion project on github

Contact me!

- Mail: David.Truyens@tddatech.be
- Twitter: @davidtruyens
- Github: https://github.com/DavidTruyens
- Fusion model: http://a360.co/1XD0Xt4