Simulated Plant Approach

LIGO Caltech

Koji Arai, Jameson Rollins,
Joe Betzwieser, Rana Adhikari

2016 March LVC Meeting

LIGO DCC G1600511

"Simulated Plant"

- is an IFO emulator for digital control system

- is realized by realtime codes of the digital control system itself

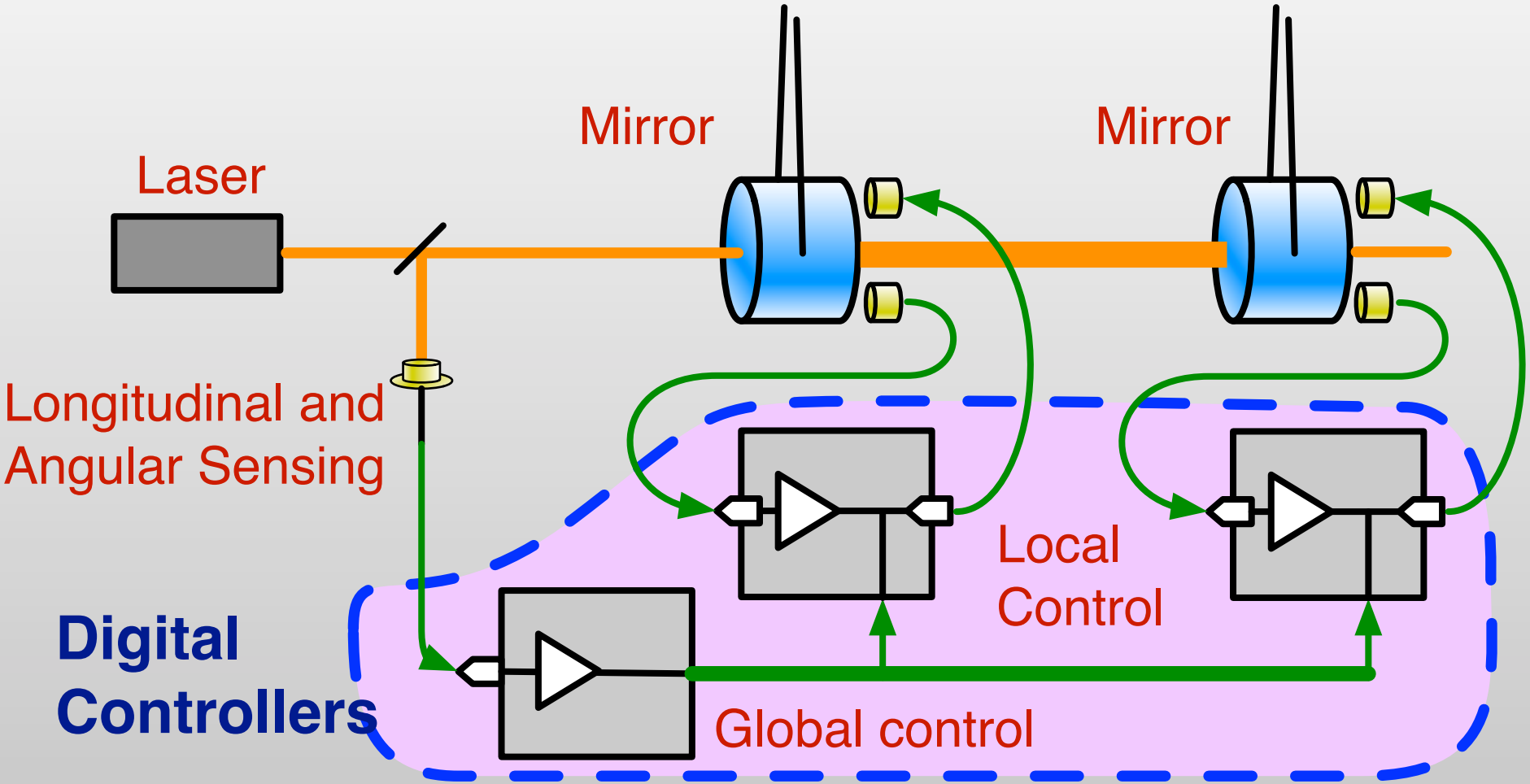
- imitates responses of interferometer components

What is this?
Why do we need this?
How can we make it?
What has been done so far?

Basic Idea

Simplified view of the interferometer control:

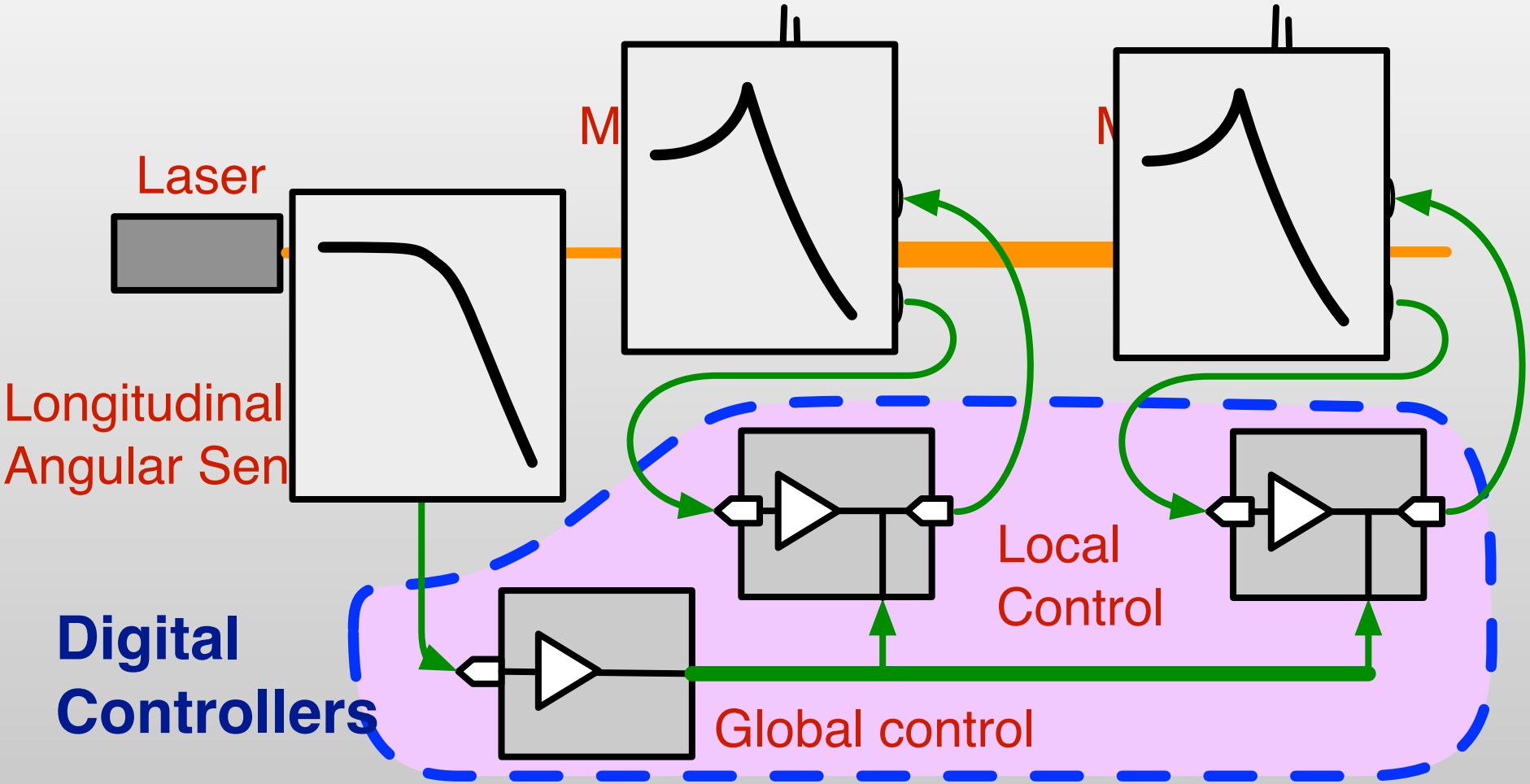
Local control (suspension, seismic platform)
+ Global control (interferometer)



Basic Idea

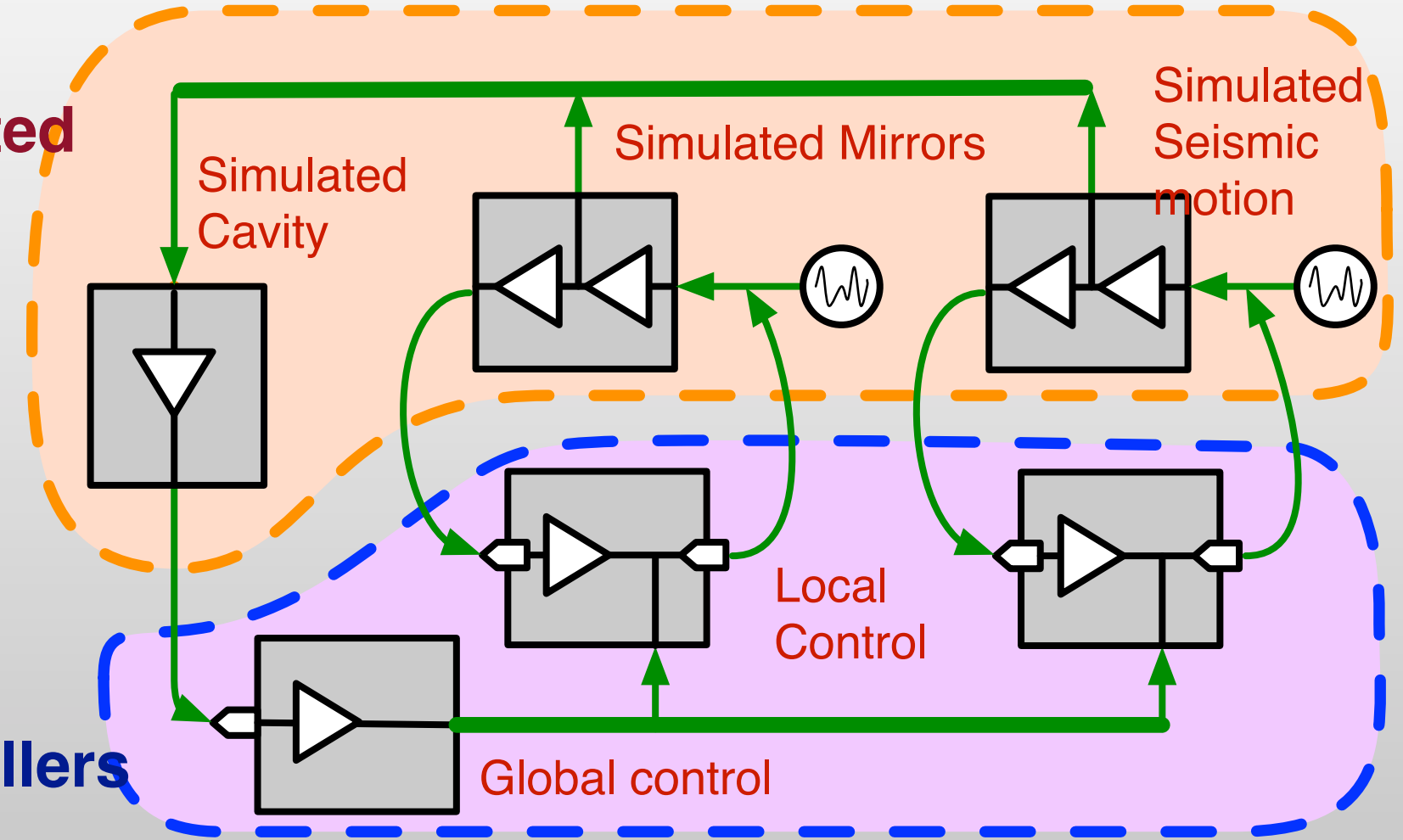
Simplified view of the interferometer control:

Local control (suspension, seismic platform)
+ Global control (interferometer)



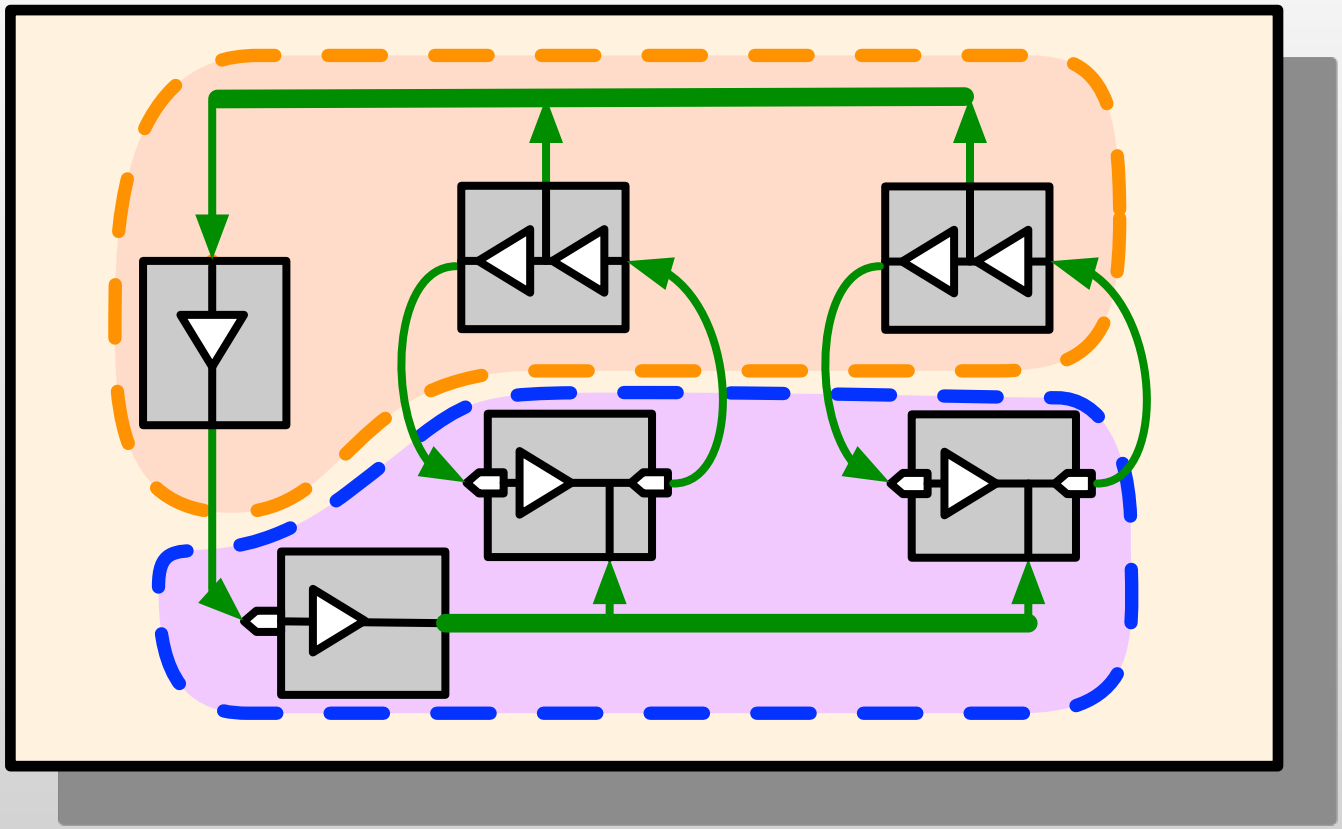
Basic Idea

Replace the hardware responses with digital filters
==> simulated plants
The controllers do not notice the swap



Basic Idea

The entire IFO/control could be realized on a single or small number of machines

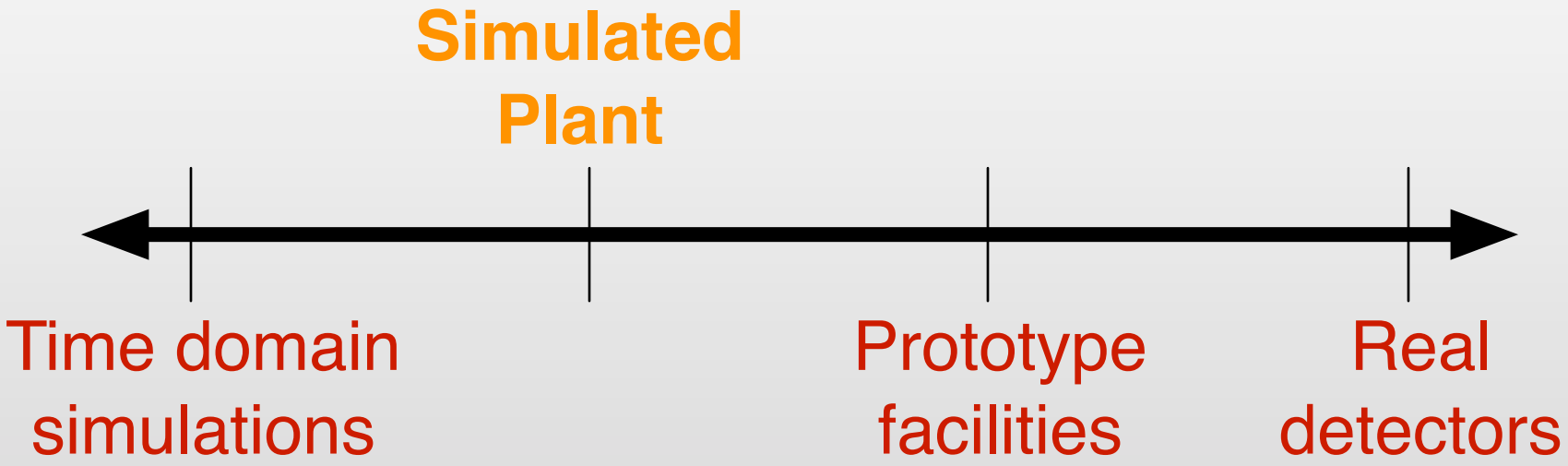


Simulated Plant version of LIGO IFO and Control

Comparison with other tools

Simulated plant ≠ a precise time-domain simulator

Limitations: computational cost, sampling rate, ...



rather like a development environment for IFO controls

=> fill the gap between the simulations and IFOs

Benefit (1)

- Separate software issues from IFO issues

Software R&D / initial test without using IFO time

e.g. Control models
Diagnostic tests
Scripts / Guardian codes
Screens
Measurement codes

Usual tools can be used (e.g. dtt, dataviewer, NDS2, ...)

IFO time is getting more precious

Let's minimize downtime for code implementation

Benefit (2)

- Designing / tesing control system for 3G detectors
  before the hardware becomes ready

  Reflect experiences with the emulator to the design

  Save eventual commissioning time

Benefit (3)

- Enables offline study on implementation of complex / advanced control systems

... may require custom C codes
- Adaptive control
- Modern control

... may require many repeated trials
- Machine learning
- Genetic algorithm

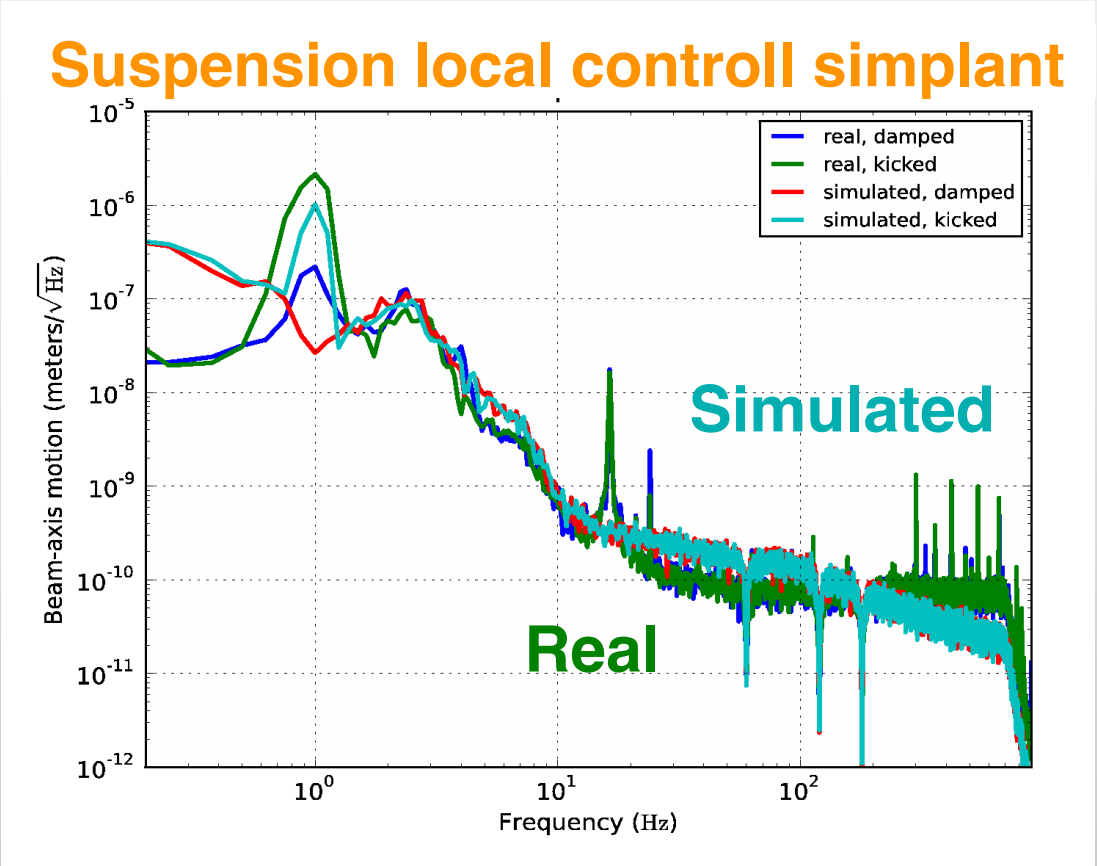
=> multiple accelerated IFO emulation?

Simulated Plant Realization

- Examples:

Damping control of a suspension system
(LIGO-G1100590, Betzwieser J., Rollins J.G., et al.)

Control model of HEPI / HAM-ISI
(LIGO-G1300850, Castillo J, Betzwieser J.)

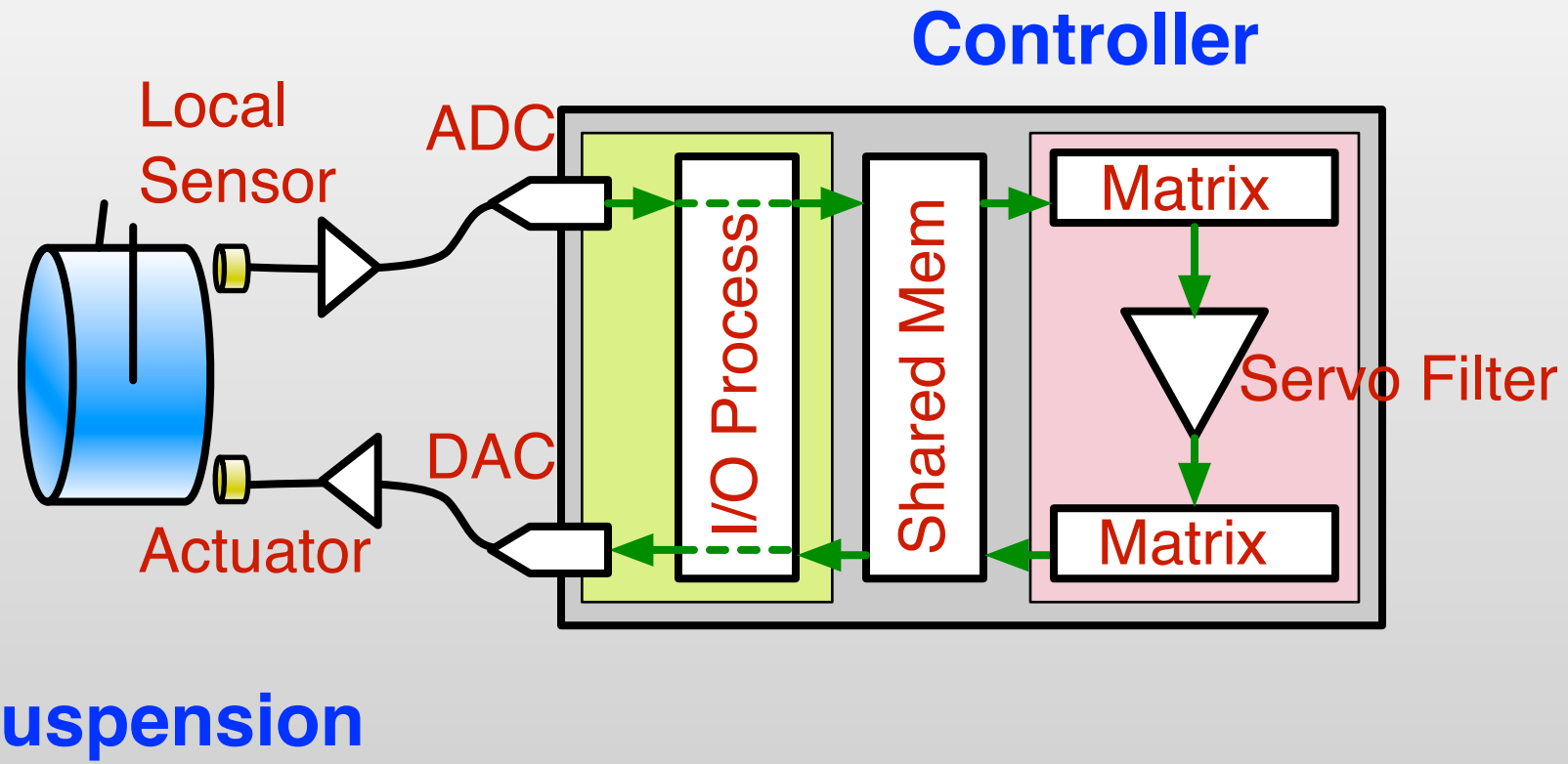


Simulated Plant Realization

- How to realize SP?

LIGO CDS (Control and Data System)

e.g. Suspension Local control

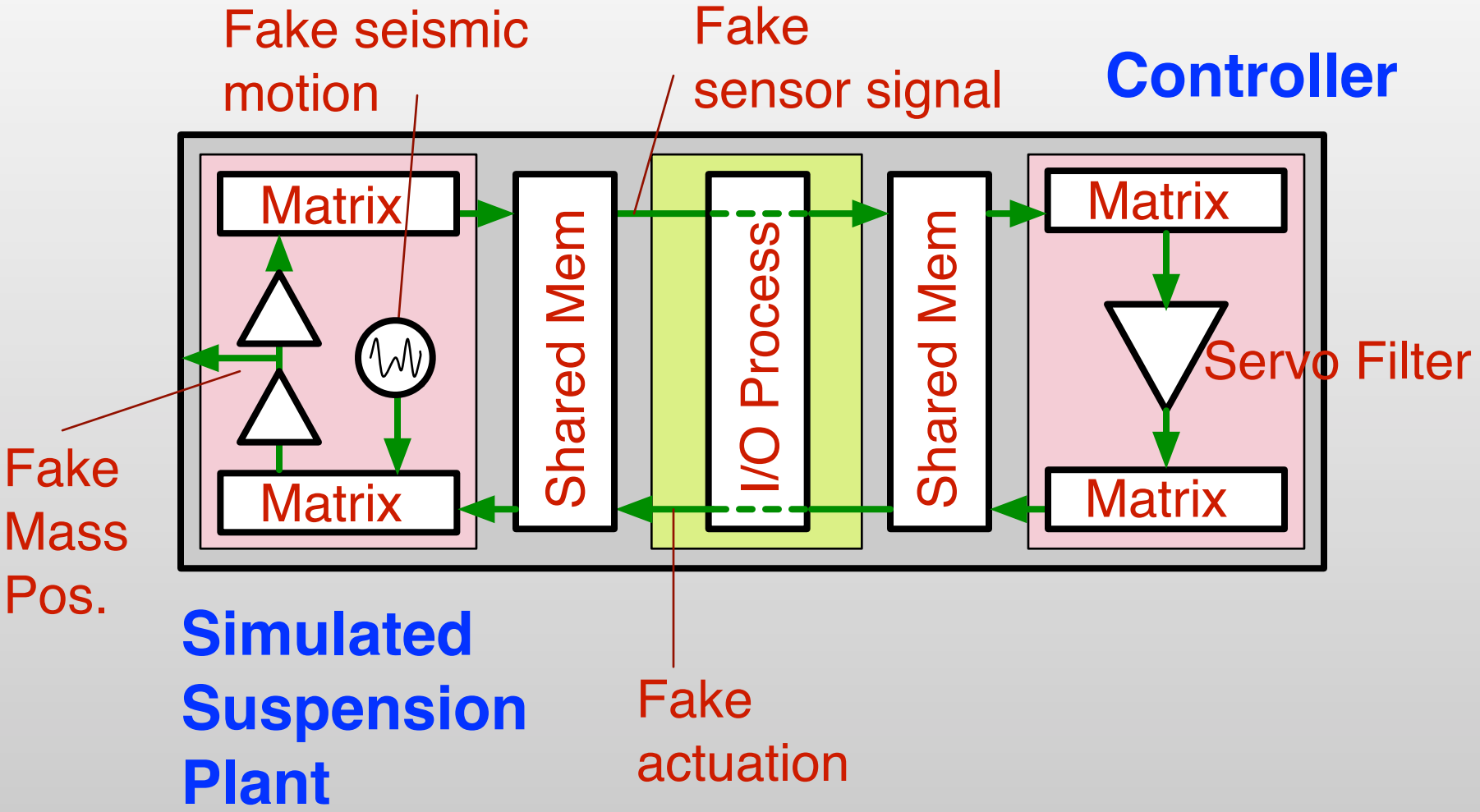


Simulated Plant Realization

- How to realize SP?

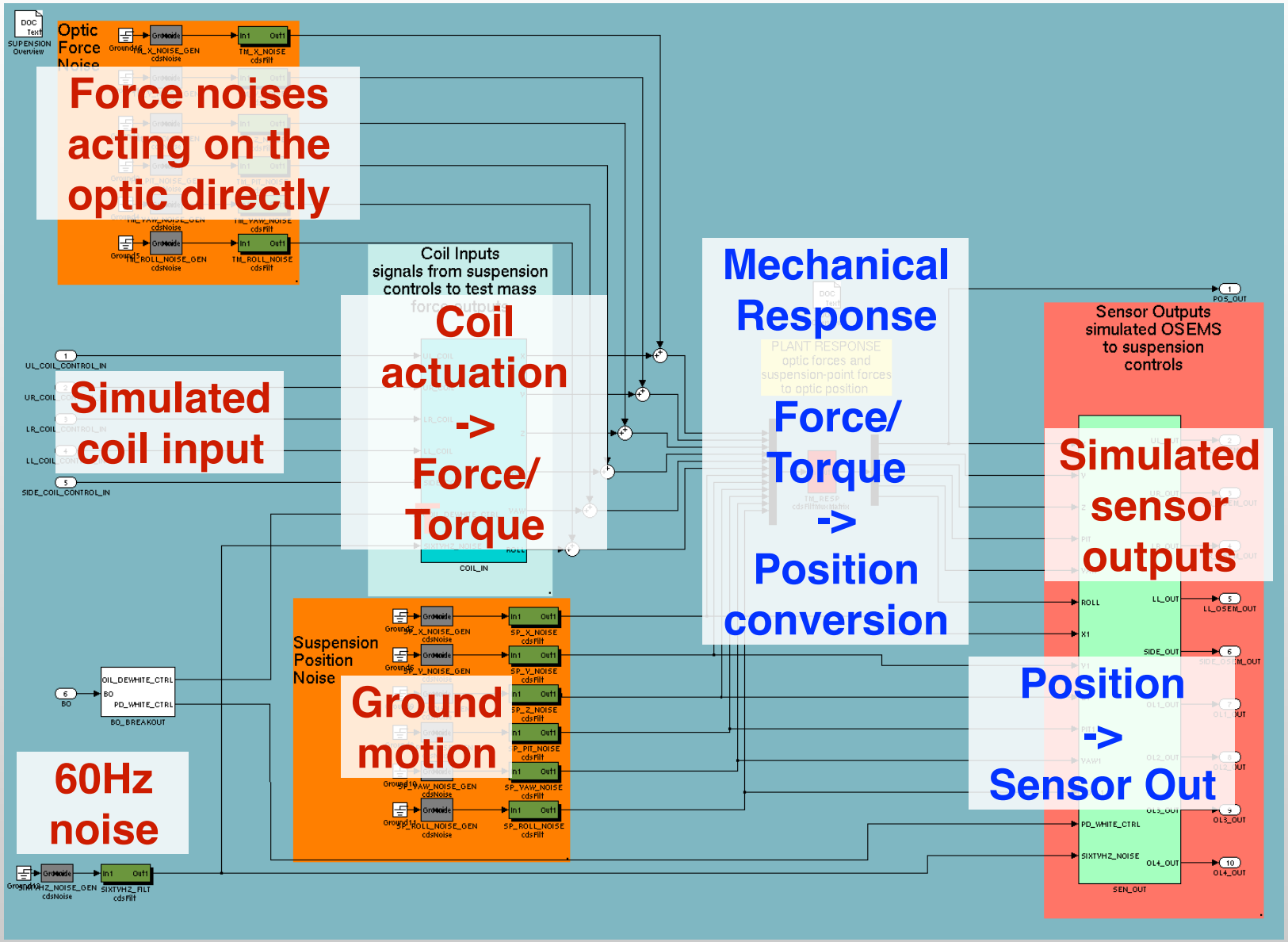
LIGO CDS (Control and Data System)

Suspension Sim-Planr / Controller



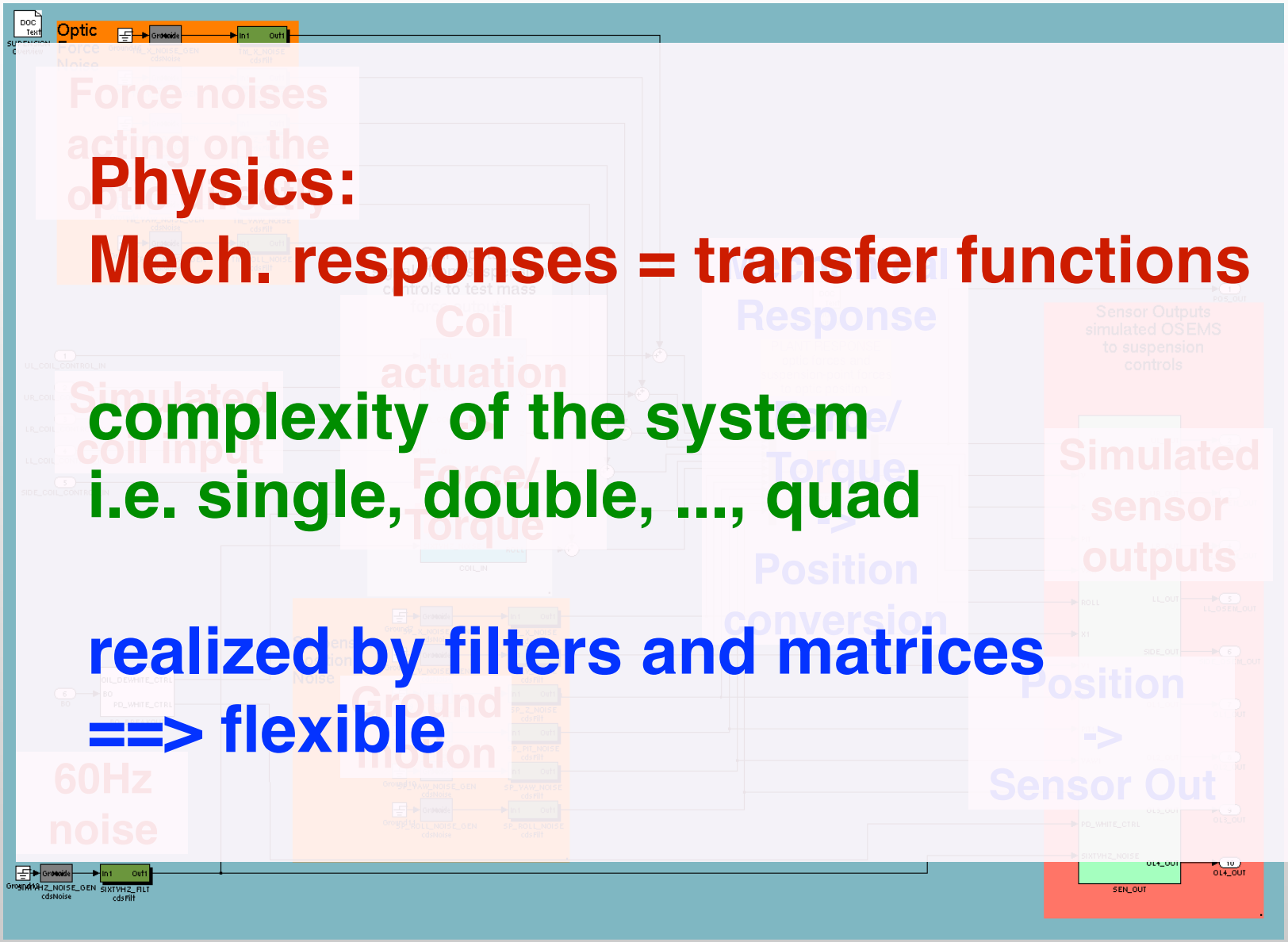
Suspension Sim-plant

Formed by matrices and arrays(or matrices) of filter modules



Suspension Sim-plant

Formed by matrices and arrays(or matrices) of filter modules



Physics:
Mech. responses = transfer functions

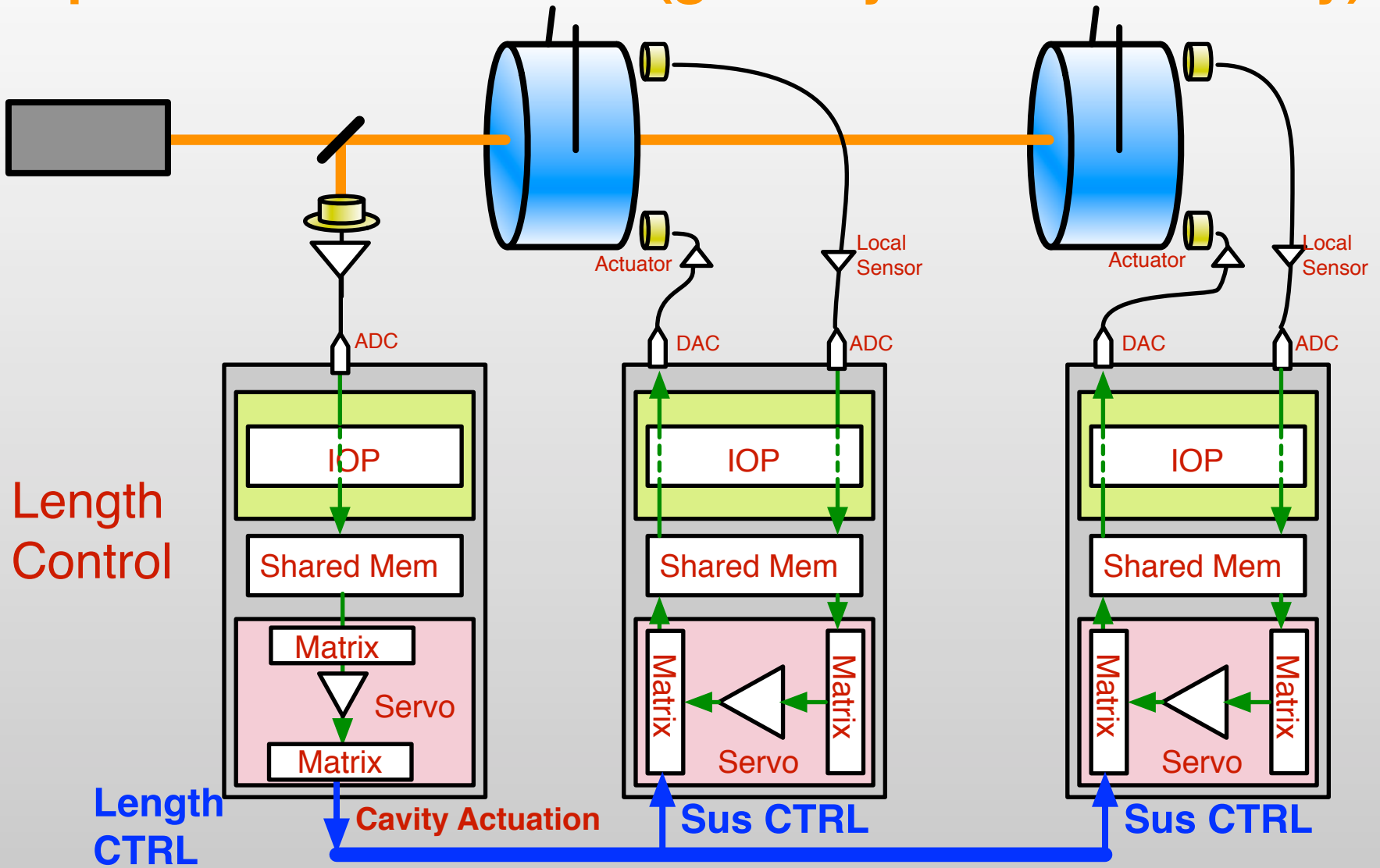
complexity of the system
i.e. single, double, ..., quad

realized by filters and matrices
==> flexible

Global sim-plant

global control ~ Multiple machine case

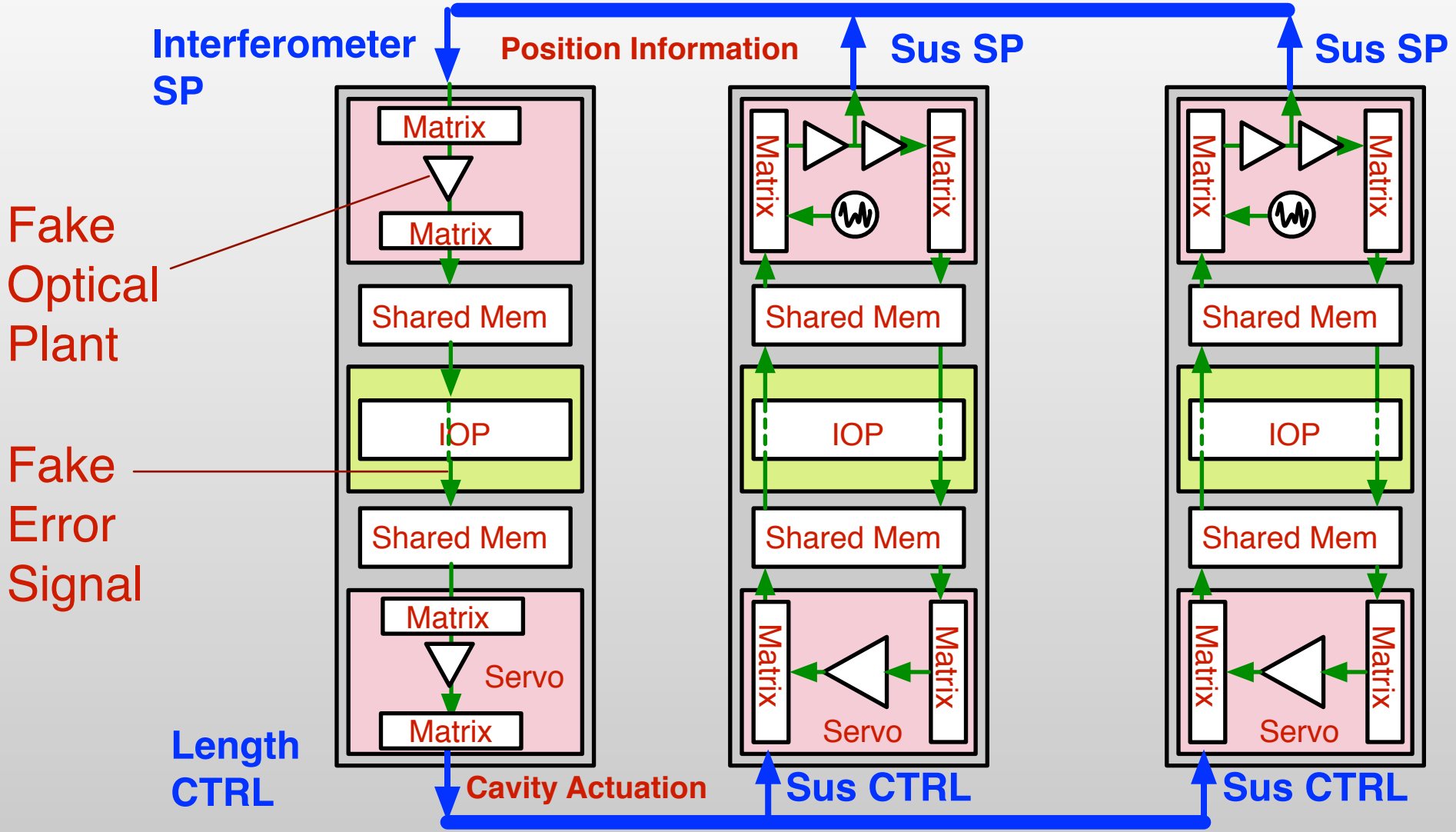
RFM/DolphIn communication (globally-shared memory)



Global sim-plant

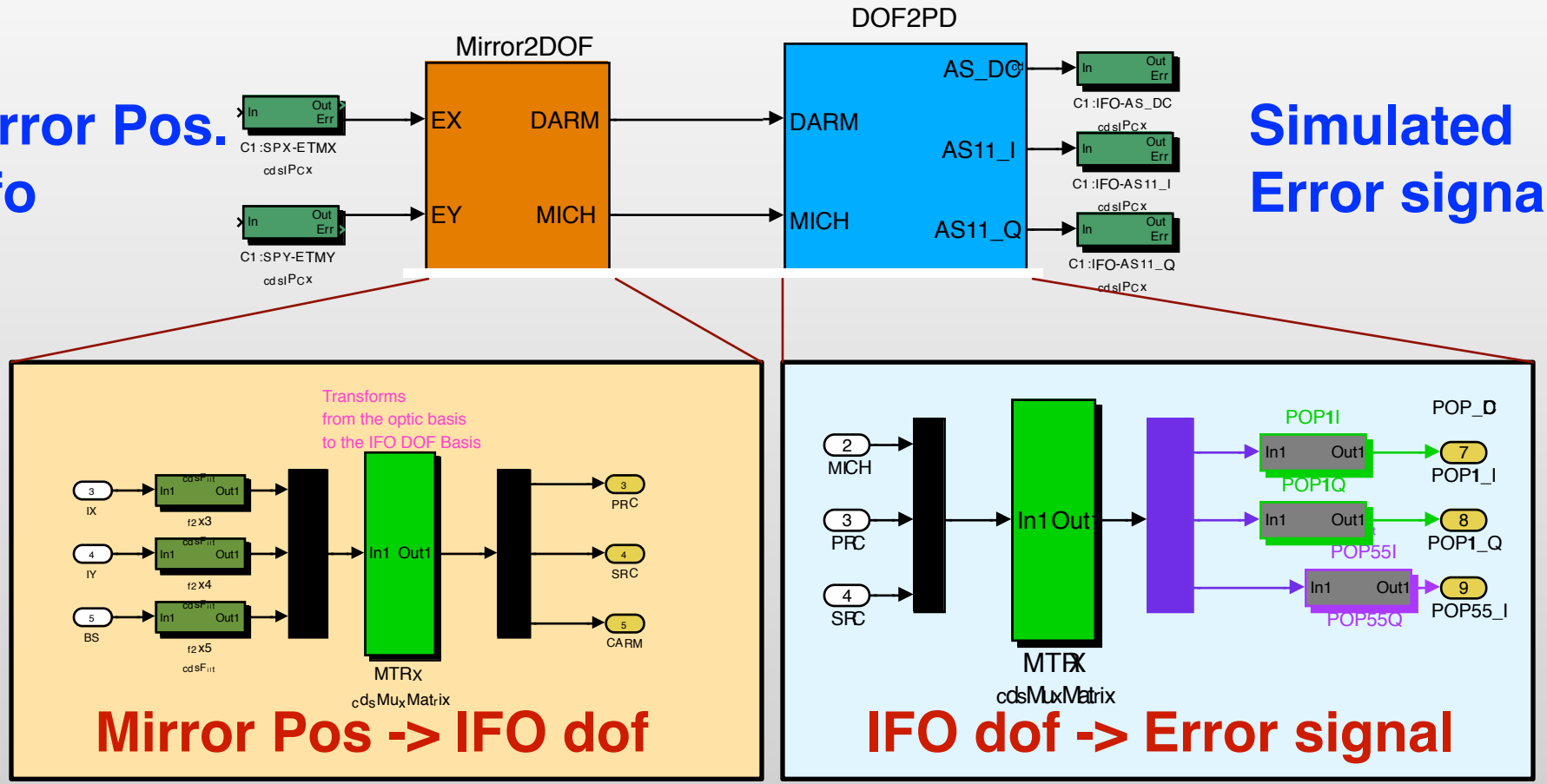
Length Controller ~ Multiple machine case

RFM/PCIe communication (globally-shared memory)



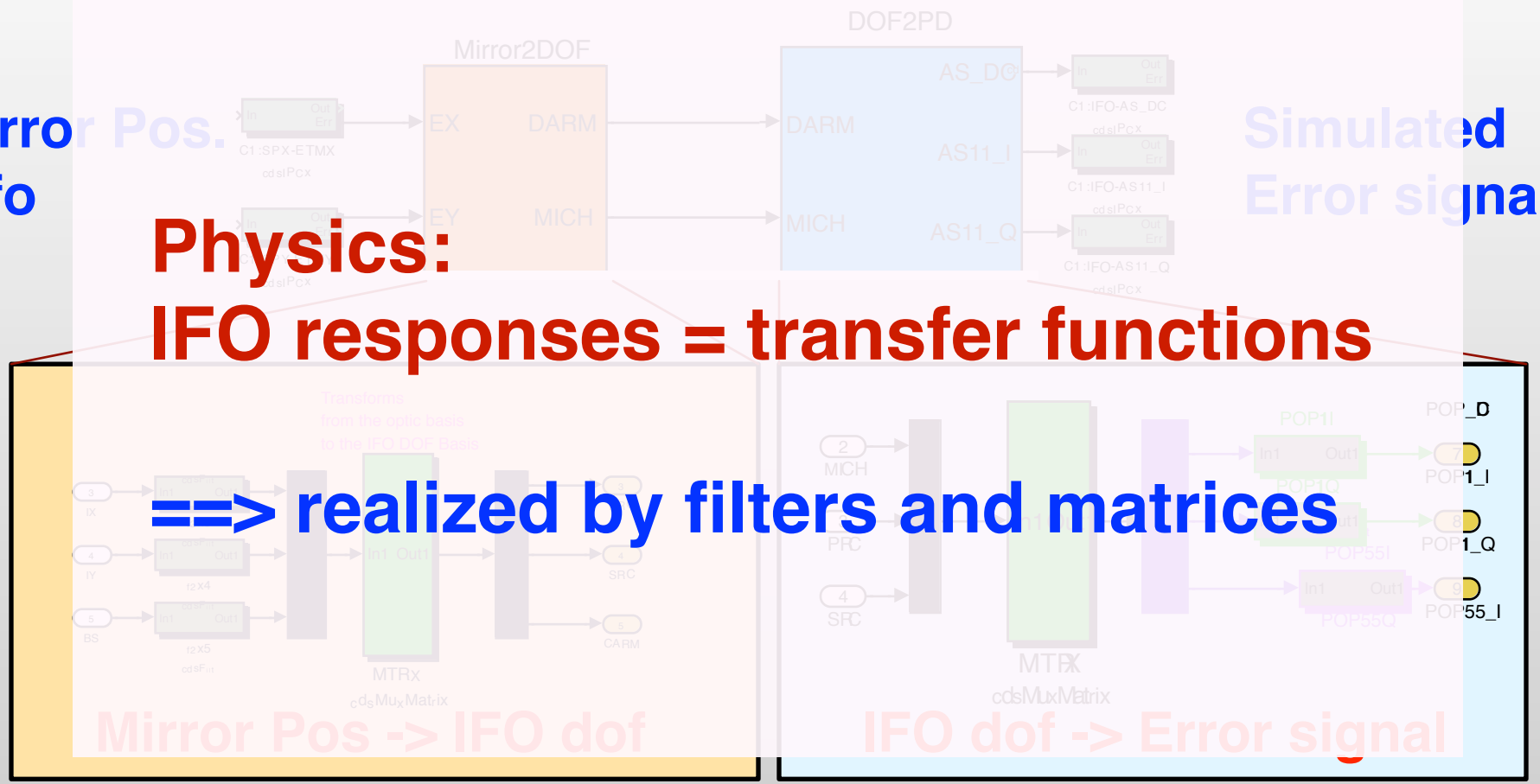
IFO Simulated Plant

Formed by matrices and arrays(or matrices) of filter modules



IFO Simulated Plant

Formed by matrices and arrays(or matrices) of filter modules



Some thoughts

- We want a standalone CDS
  with no ADC/DAC/timing hardware

- Can CDS codes be run at an accelerated rate?

- How to realize nonlinear responses of subsystems?

- How to implement IFO responses
  Simple filters are already useful at the beginning
  How much details can we implement?

Summary

- Simulated plant: an IFO emulator
  - Realized by the digital control system itself
  - Offers environment for control development

- Run an IFO and its control system on software
  - Releases IFO time from SW development
  - Enables us to work on
    control system design
    complex / advanced control problems
    without having the real detector
  - Possibility to have multiple / accelerated IFOs

- Progress & Future
  - Proof of concept test with local/global control
  - Next: work on more complicated control examples

Spare slides

Benefit (3)

- Enables offline study on implementation of complex / advanced control systems

... may require custom C codes

Adaptive control

Modern control

... may require

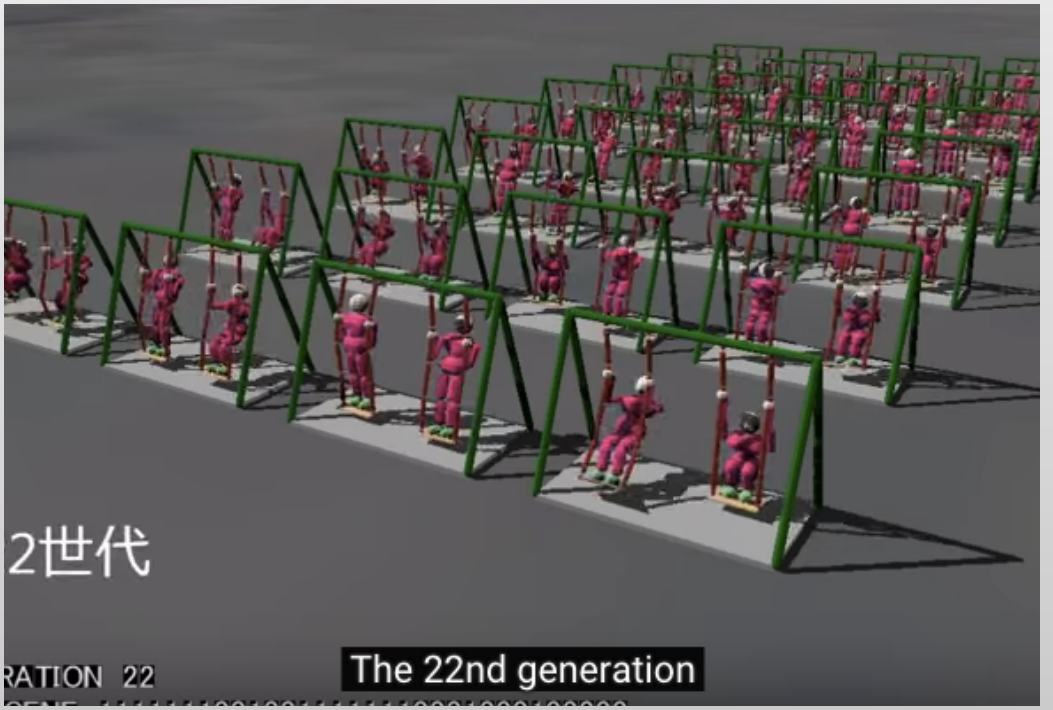
many repeated trials

Machine learning

Genetic algorithm

=> multiple accelerated IFO emulation?

Learning how to swing with genetic algorithm (x64)
https://youtu.be/w1MF0Iz0p40



Simulated Satellite

Similar system is already used for satellite operation