

# Промышленные киберфизические системы

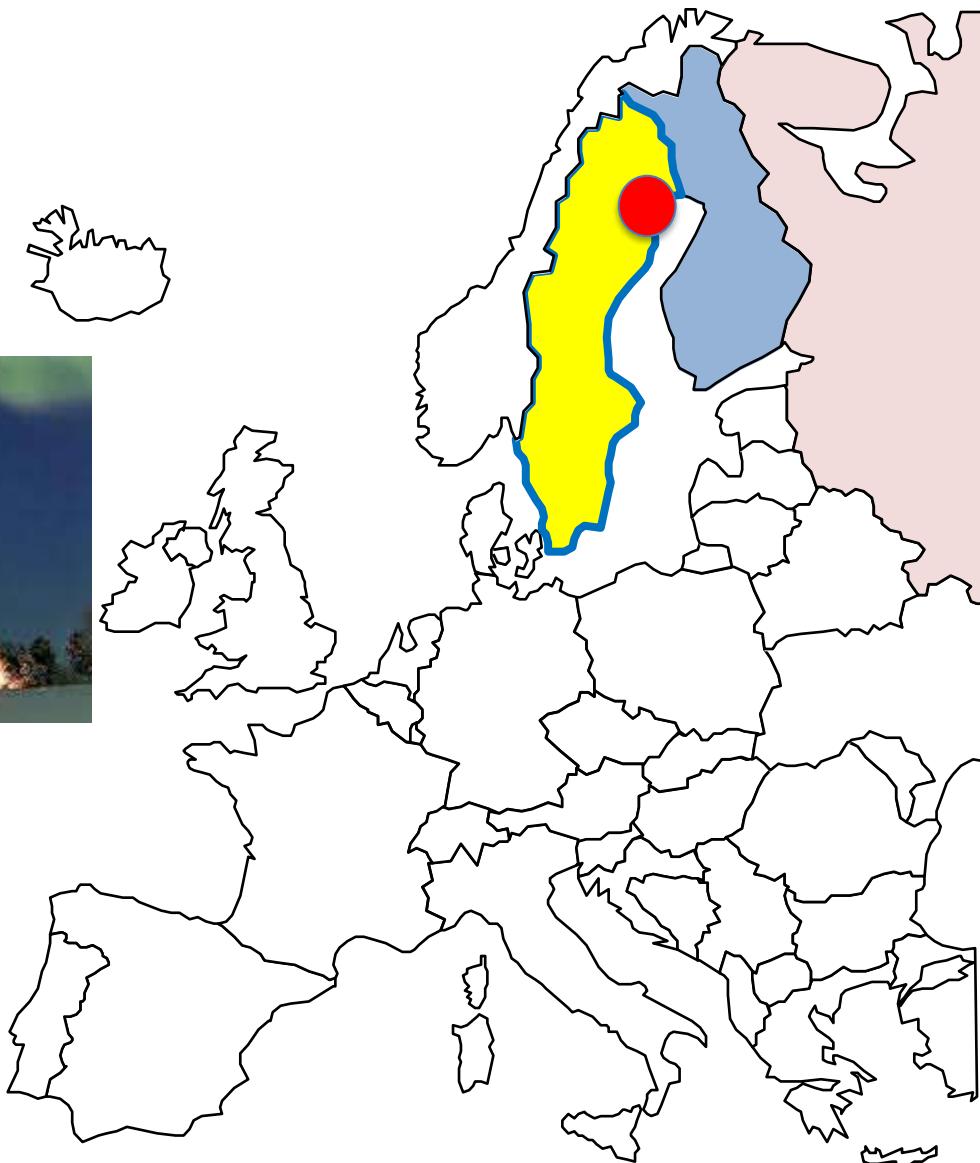
## Industrial Cyber Physical Systems

Валерий Вяткин  
17 августа 2020 г.

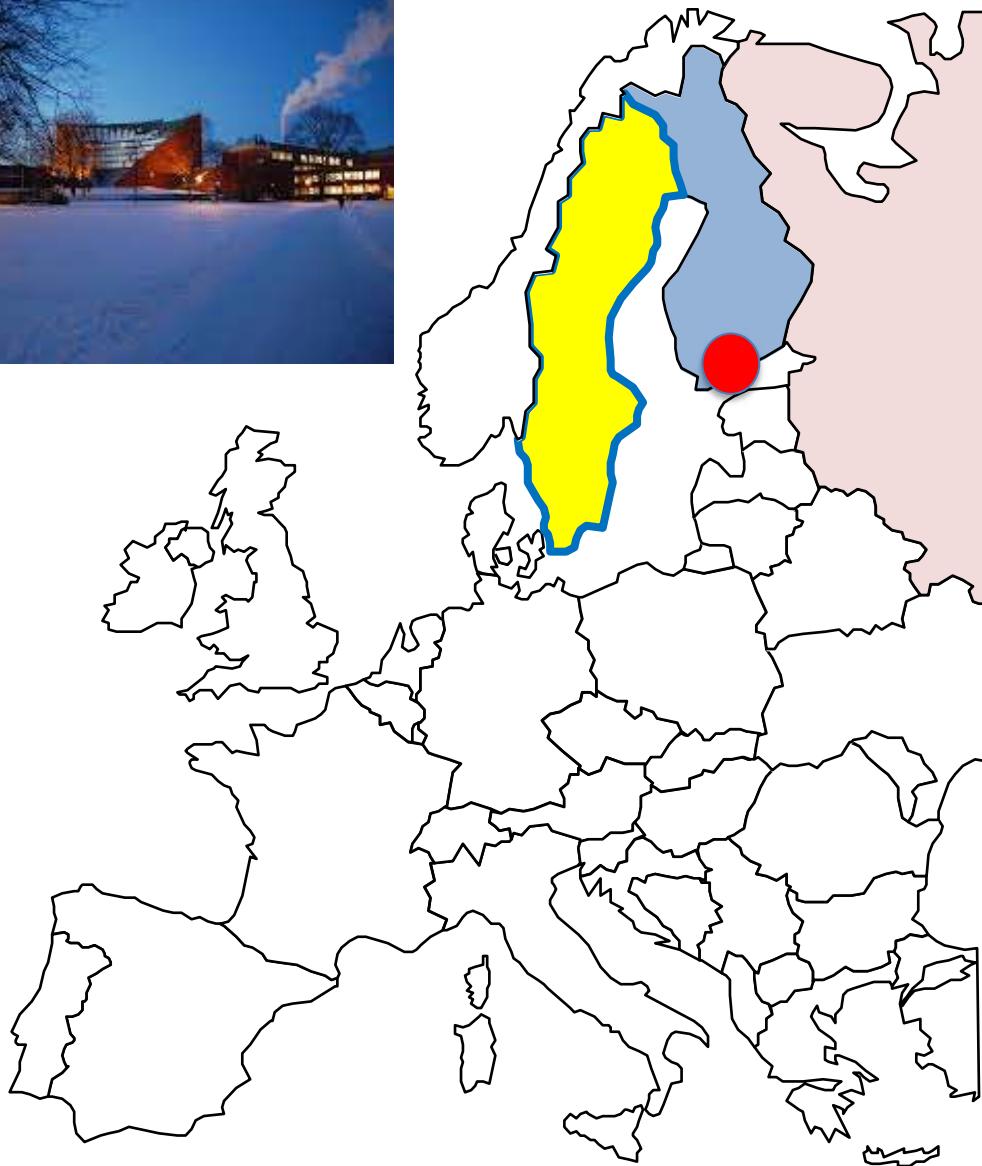
[Valeriy.Vyatkin@gmail.com](mailto:Valeriy.Vyatkin@gmail.com)



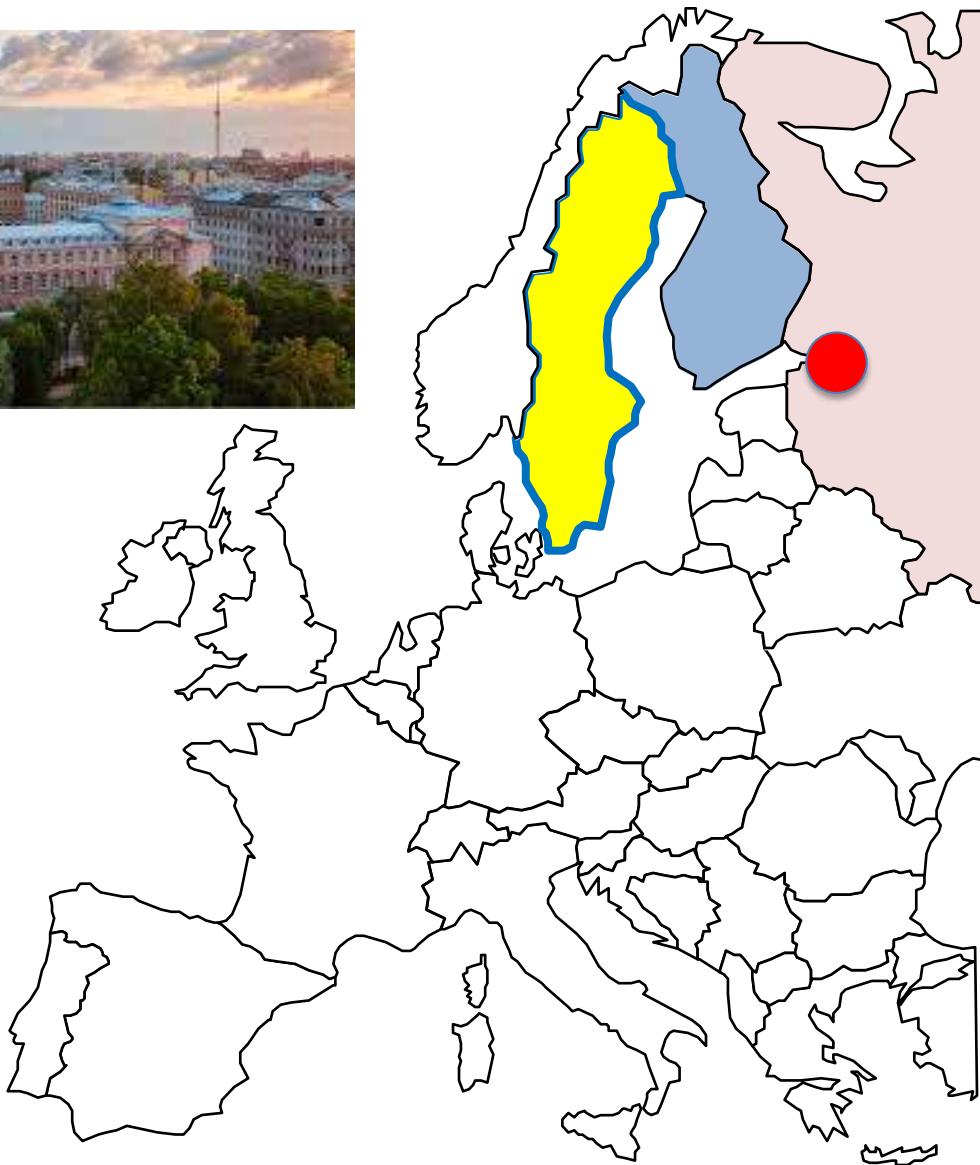
# Luleå University of Technology



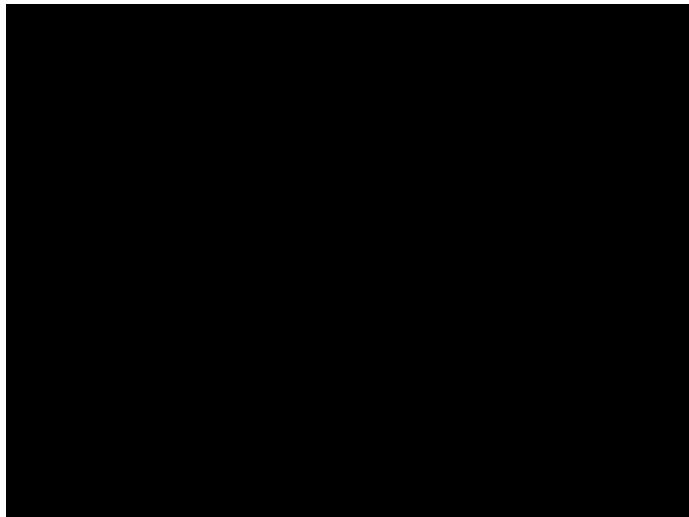
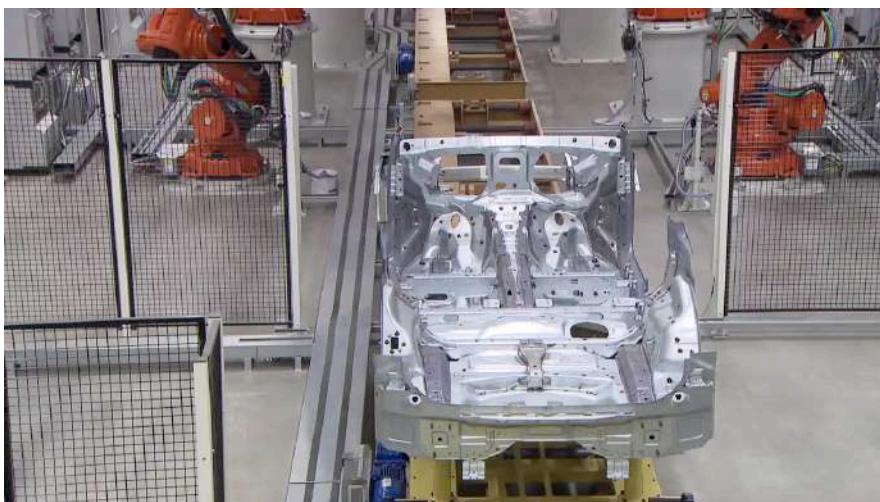
# Aalto University



# Университет ИТМО



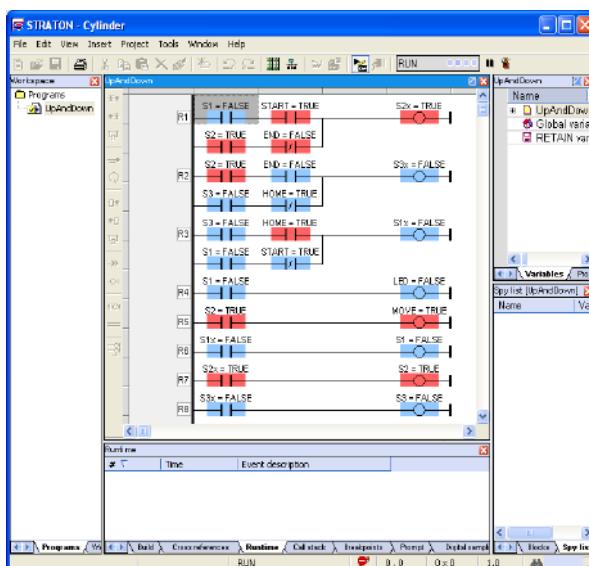
# Промышленная автоматизация



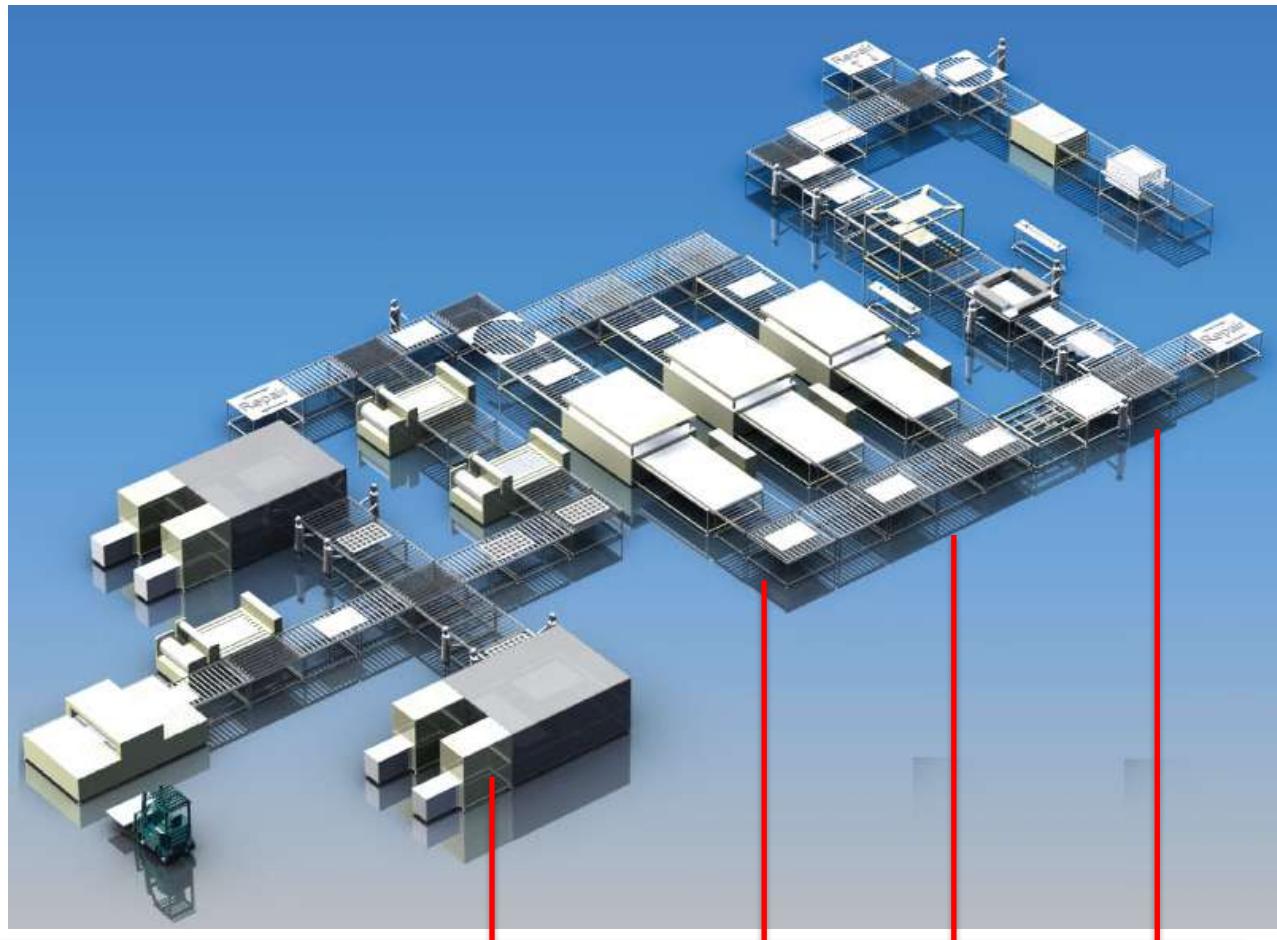
# Plan of the lecture

- Programmable automation
- Computation model
- CPS
- Flexible reconfigurable systems
- New languages for distributed systems
- Challenges of distributed decision making
- Multi-agent systems
- Towards plug and play

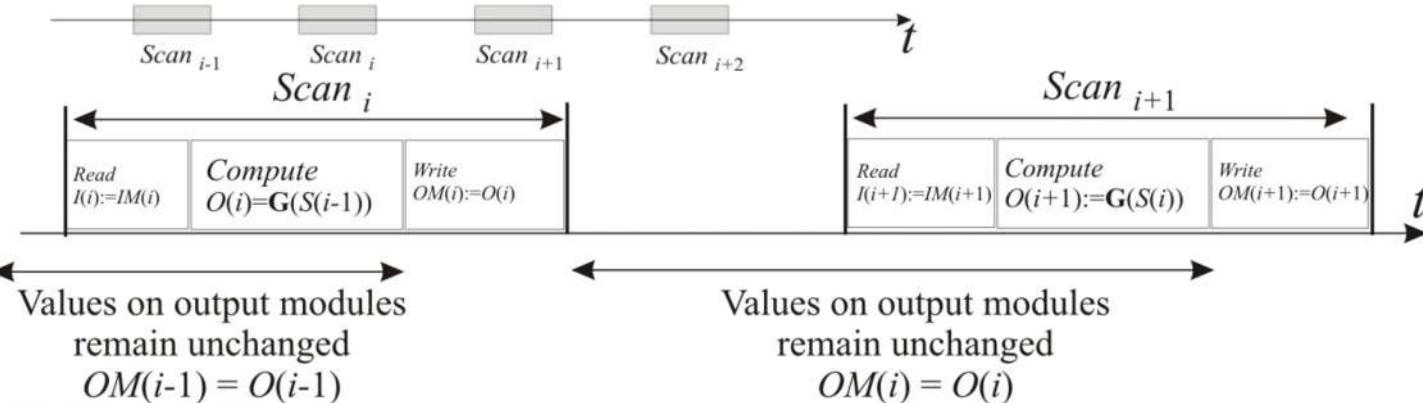
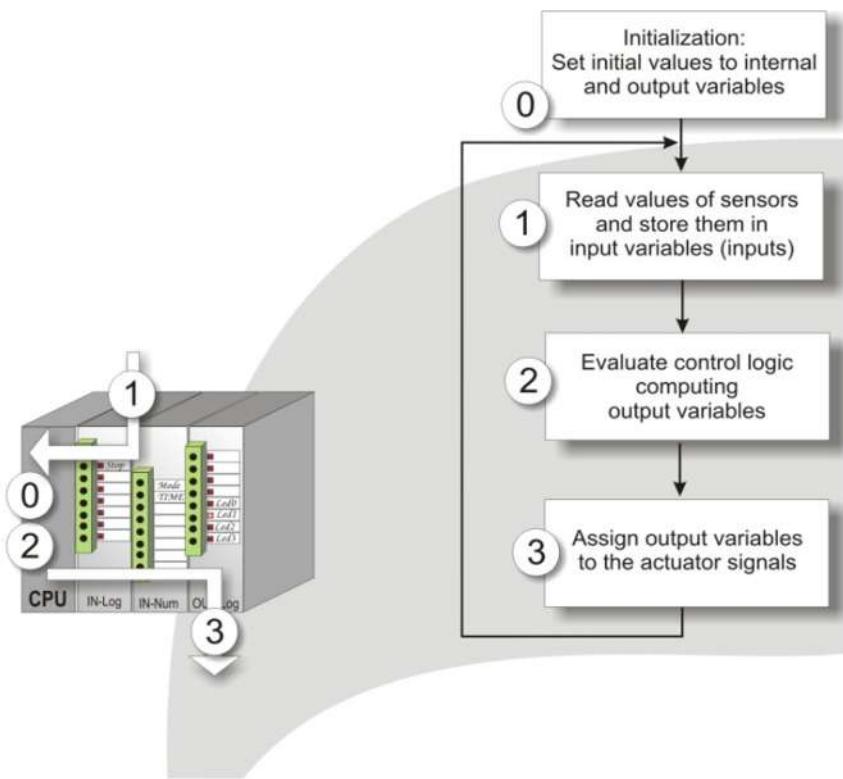
# Программируемая автоматизация сегодня



PLC



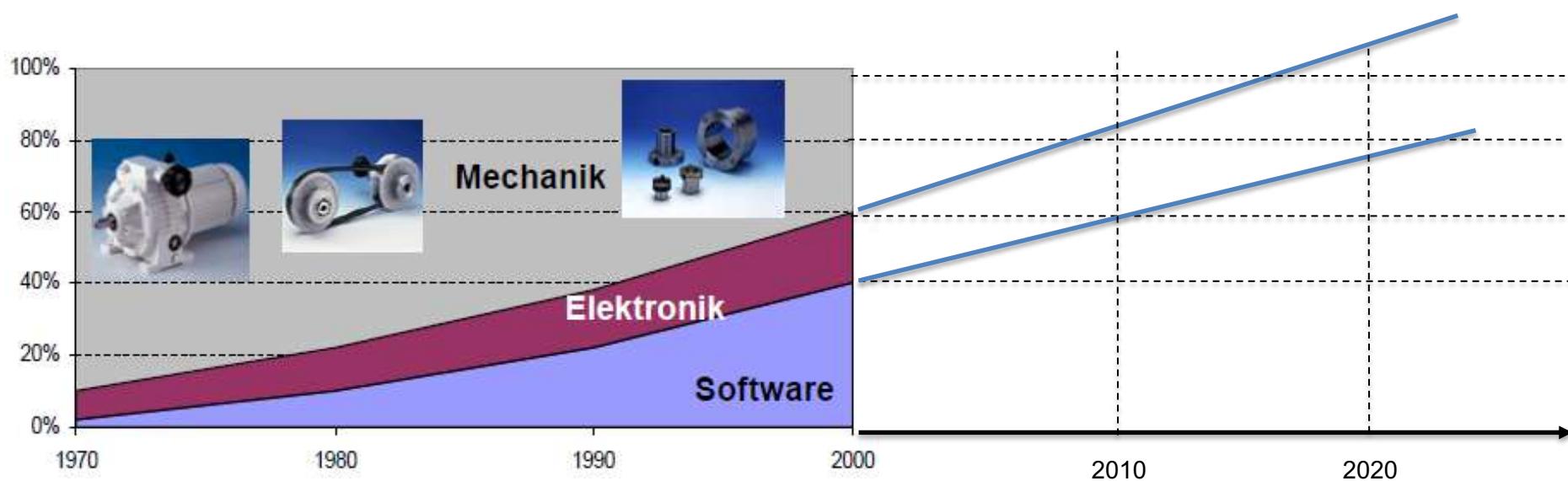
# Синхронная модель вычислений



## Свойства:

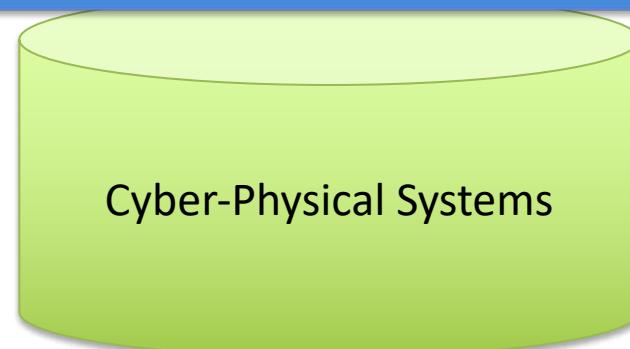
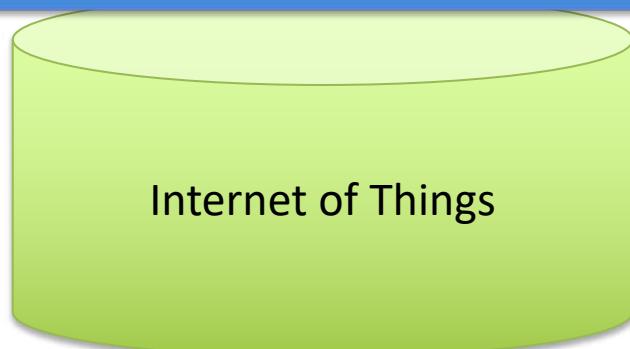
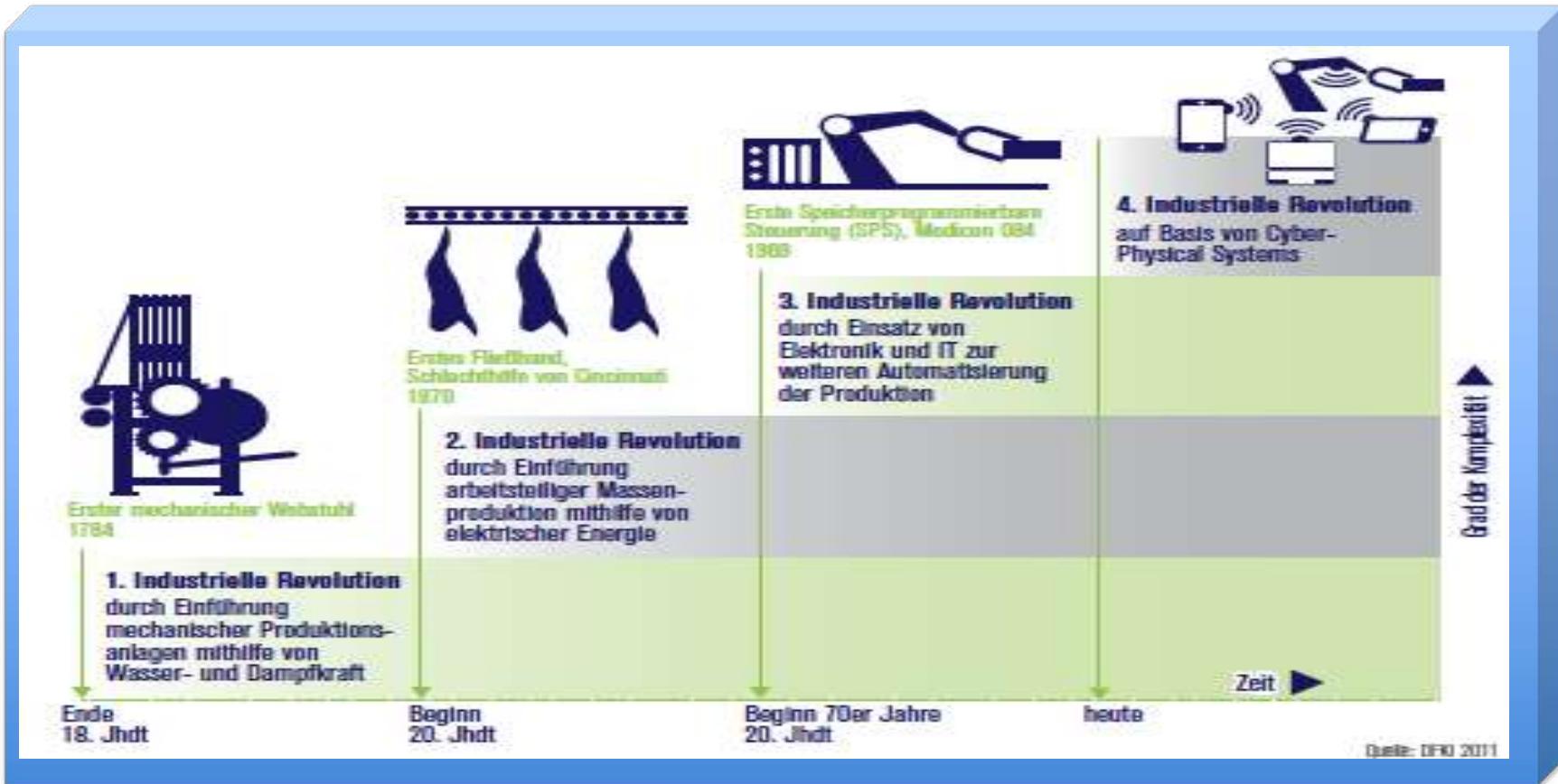
- Детерминированность – выдаёт одни и те же выходные сигналы, при тех же входных данных
- Надёжность – в случае ошибки датчика, быстро восстанавливает правильный результат
- Время скана должно быть коротким!

# Доля программного обеспечения в промышленной автоматизации



According to Verband Deutscher Maschinen- und Anlagenbau e.V. (VDMA)

# Industrie 4.0



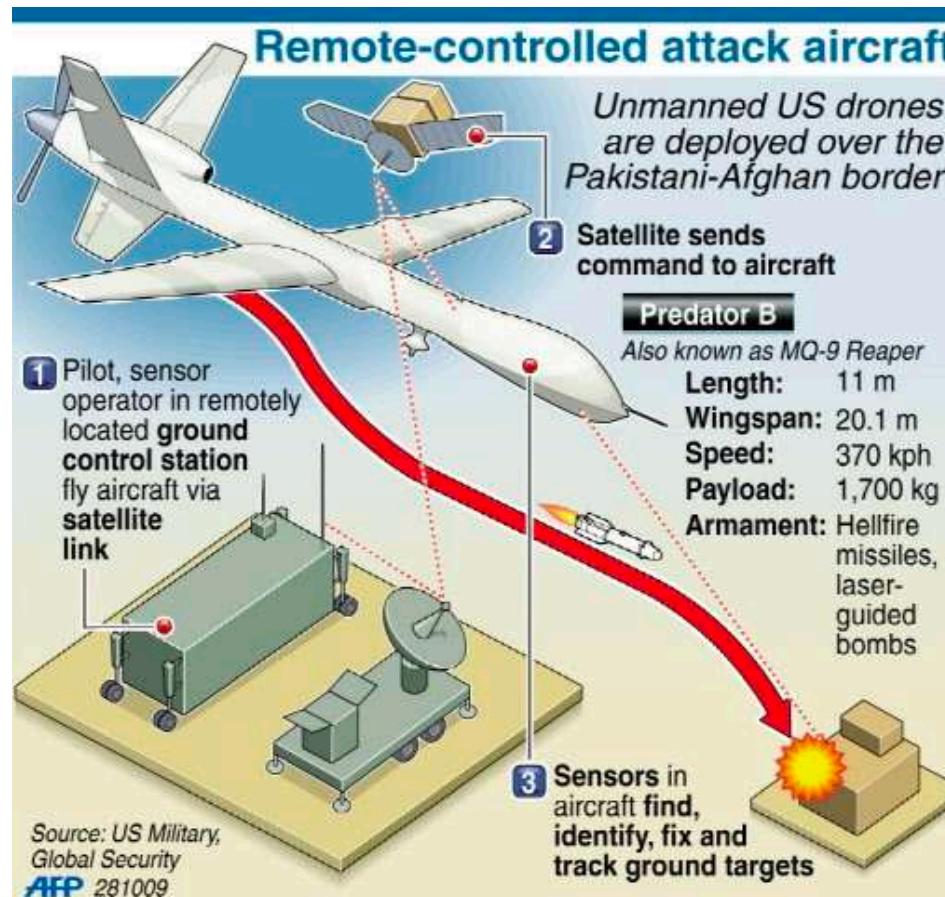
# Понятие киберфизической системы (КФС)

Киберфизическaя система (КФС) – это концепция, подразумевающая тесную интеграцию вычислительных ресурсов в физические процессы



# Причины КФС

Необходимость проектировать, моделировать и строить сложные системы со многими (десятками, сотнями, тысячами) взаимодействующих автономных узлов.



# Пример: стая дронов



# Откуда произошёл термин «киберфизические системы»?

Исследование киберфизических систем (КФС) направлено на интеграцию знаний и инженерных принципов в вычислительных и инженерных дисциплинах (сетевое взаимодействие, управление, программное обеспечение, взаимодействие с человеком, теория обучения, а также электрические, механические, химические, биомедицинские, материаловедение и другие инженерные дисциплины) с целью развития новой КФС науки и поддерживающих технологий.

Каждое представление выделяет определенные функции и игнорирует другие, чтобы сделать анализ более понятным. Как правило, определенный формализм хорошо представляет кибер или физический процесс, но не то и другое одновременно.

Baheti, Radhakisan, and Helen Gill. "Cyber-physical systems." *The impact of control technology* 12 (2011): 161-166.

# Понятие киберфизической системы (КФС)

## Отличительные особенности:

- В отличие от **встроенных систем**, модели КФС включают модель физического процесса
- В отличие от **теории управления**, модели КФС учитывают семантику реальных вычислительных систем

$$u(t) = MV(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

```
// Infinite loop.
while (1)
{
    // Get the current position.
    current_position = read_current_position();

    // Calculate the error.
    error = target_position - current_position;

    // Calculate the integral.
    integral = integral + error;

    // Calculate the derivative.
    derivative = error - last_error;

    // Calculate the Control Variable.
    pwm = (kp * error) + (ki * integral) + (kd * derivative);

    // Limit the Control Variable to within +-255.
    if (pwm > 255) pwm = 255;
    else if (pwm < -255) pwm = -255;

    // If the Control Variable is positive, run the motor clockwise.
    if (pwm > 0) motor_cw(pwm);

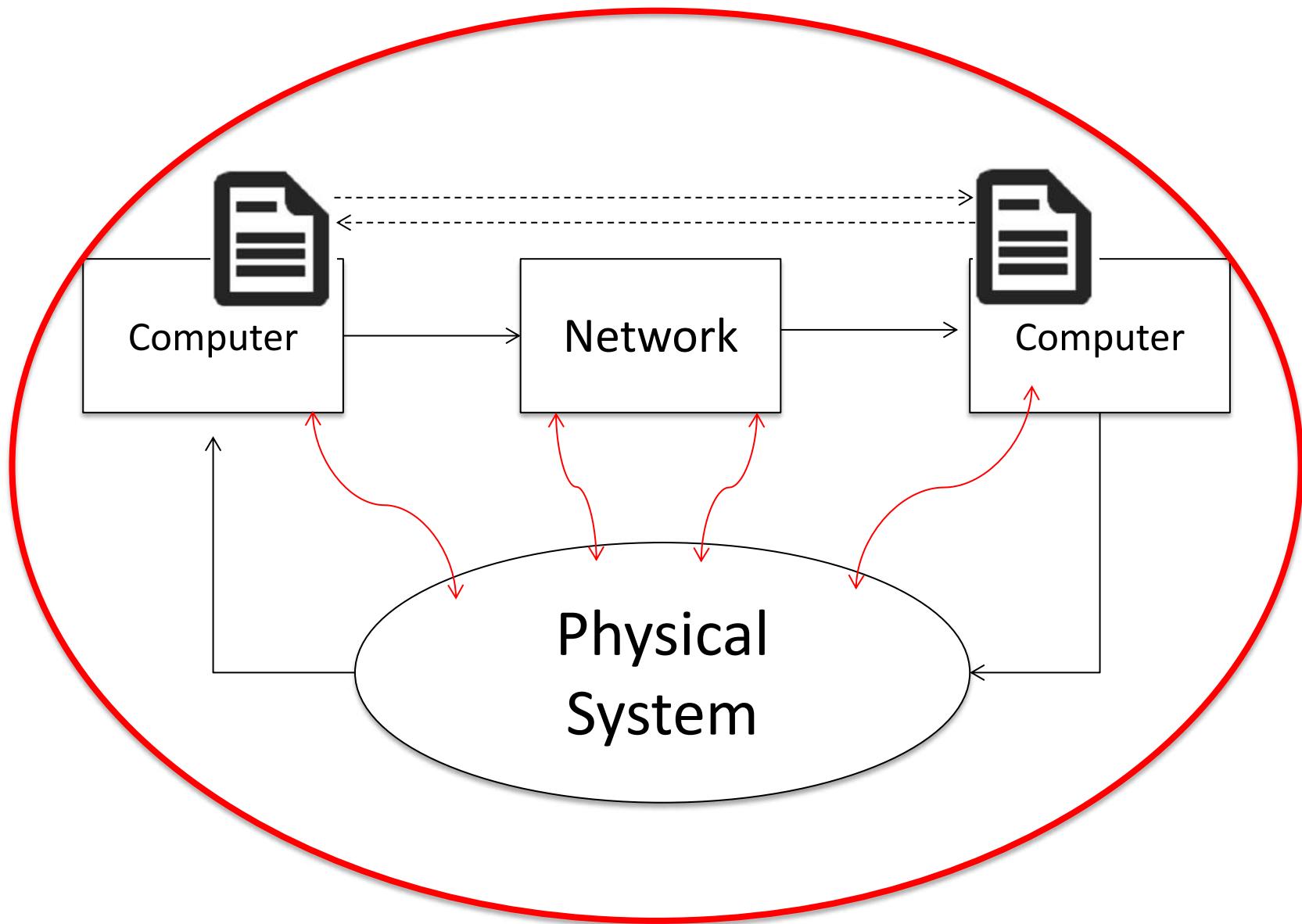
    // If the Control Variable is negative, run the motor counter clockwise.
    else if (pwm < 0) motor_ccw(-pwm);

    // If the Control Variable is zero, stop the motor.
    else motor_stop();

    // Save the current error as last error for next iteration.
    last_error = error;
}
```



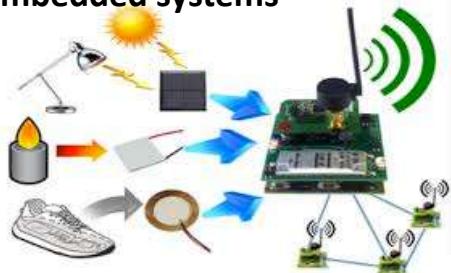
# Киберфизические взаимовлияния



# Промышленные киберфизические системы

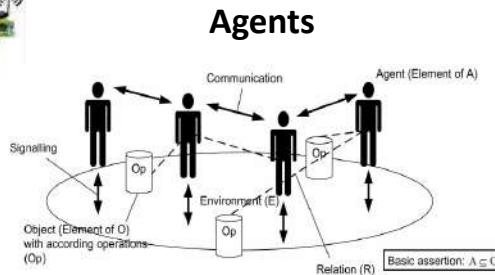
## Технологии

Communicating  
embedded systems

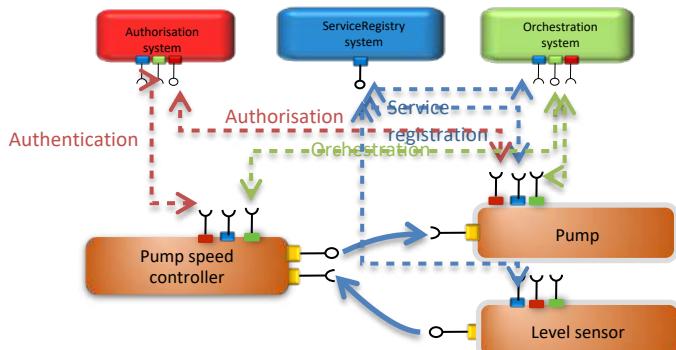


Cloud

Agents



Service-oriented architecture



## Наука КФС

Как заставить их  
работать  
предсказуемо и  
надежно?



Кибер-физическая  
инженерия

# Насколько промышленная автоматика нуждается в науке КФС?

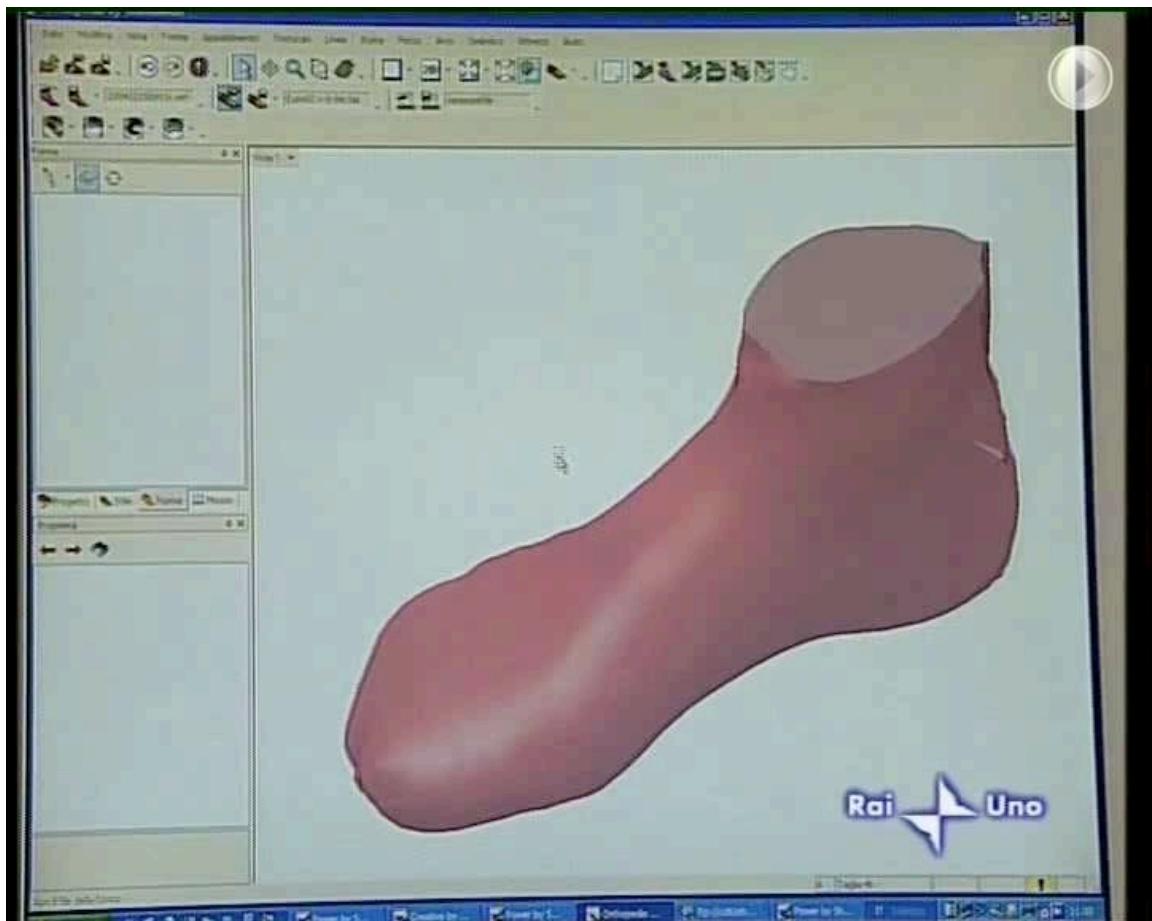
Первые очевидные мотивы:

- распределенные вычисления и
- беспроводная связь

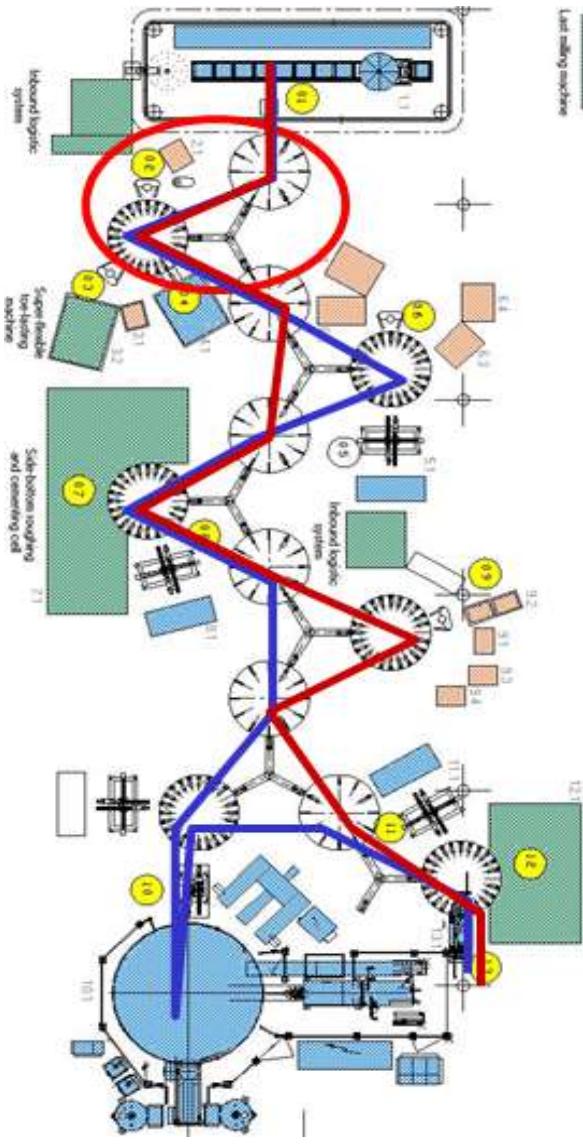
Почему беспроводная?

- Обеспечивает гибкость!

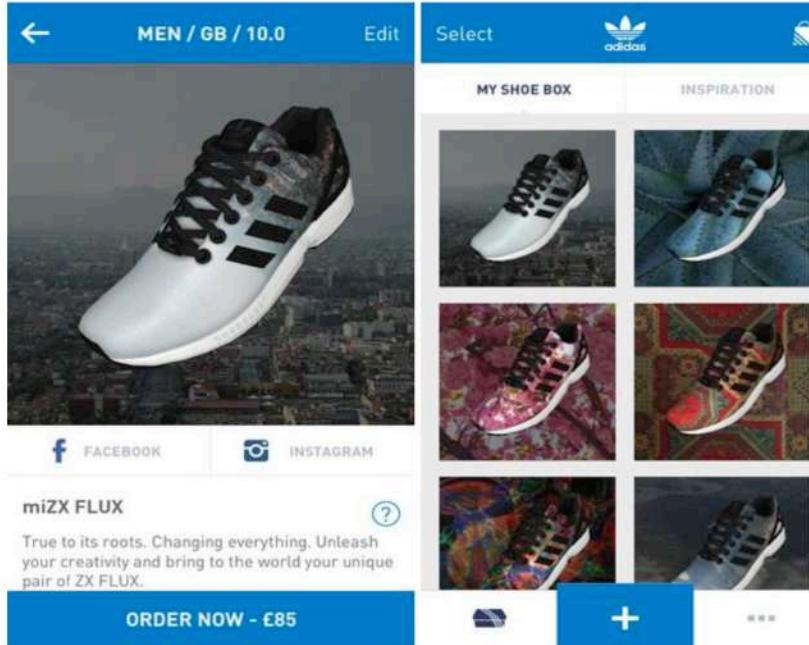
# XXI Century: Manufacturing to order!



# Flexible production line

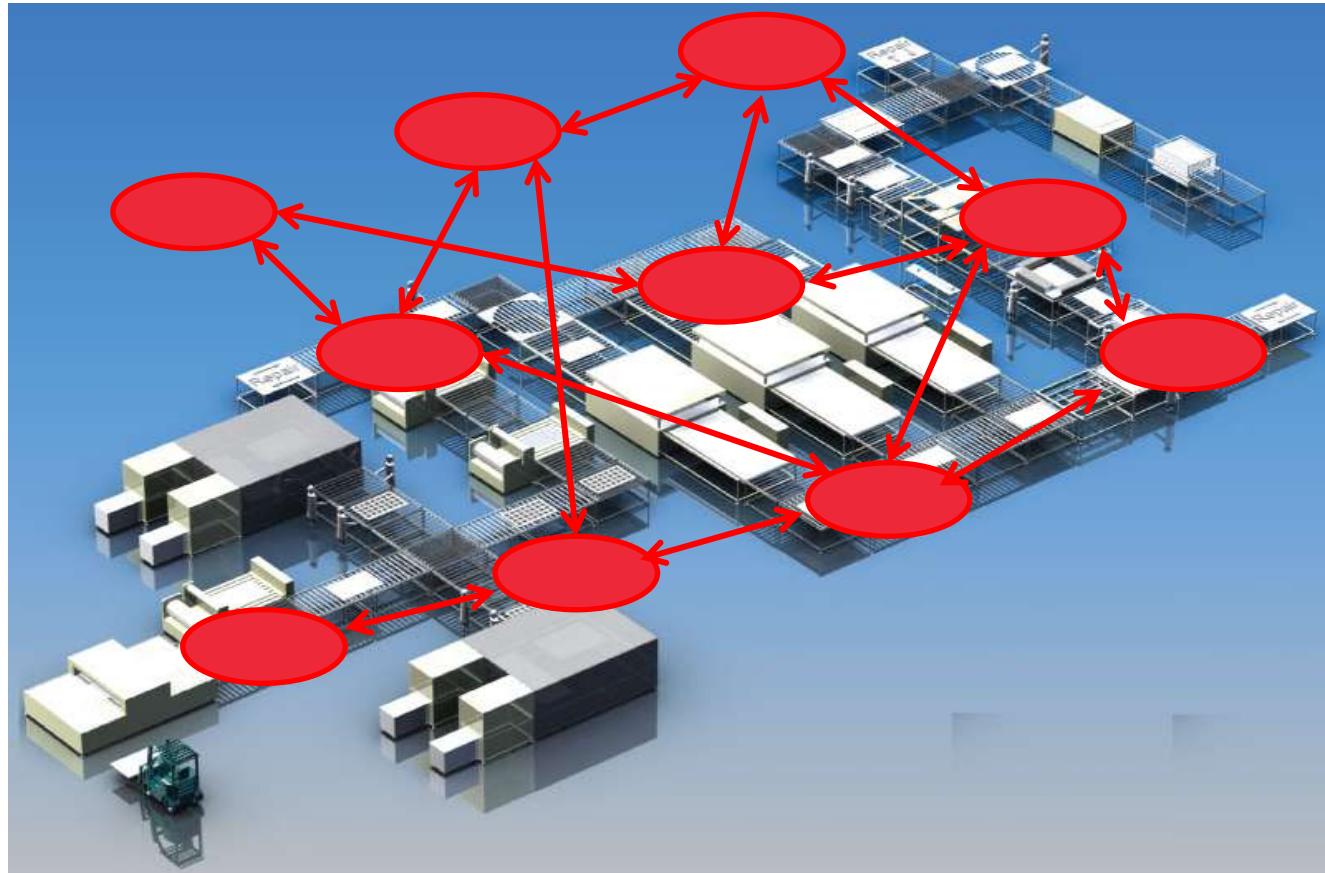


# The Adidas Speedfactory: Bringing Sports Shoes Production back to Germany by Industrie 4.0 for Mass Customization



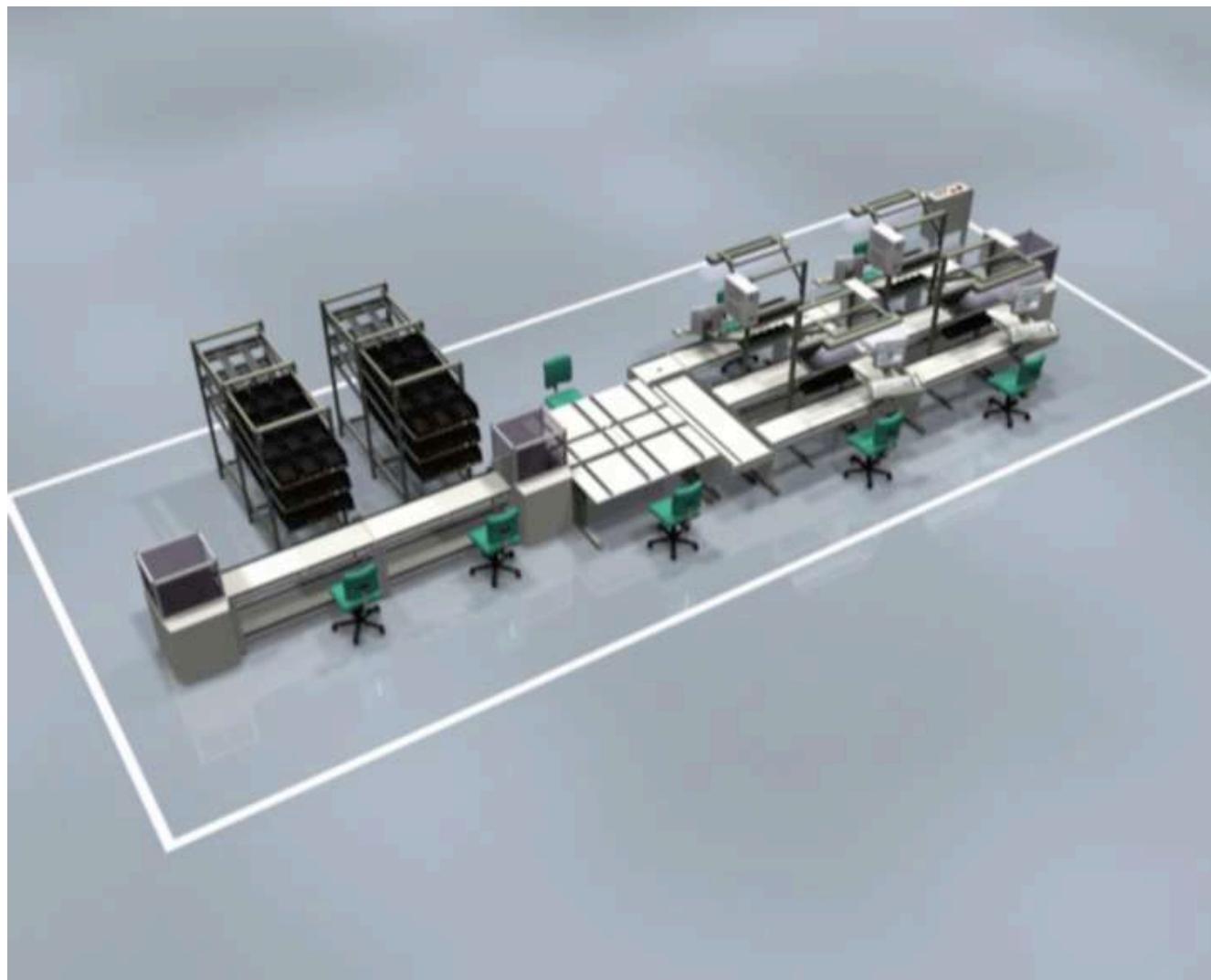
- The customers can design their own short shoes using an App.
- Since the customer wants to receive his personalized product on the next day or faster, long logistic chains from low-wage countries are no longer acceptable in the era of mass customization.
- Thus, adidas decided to open various "speedfactories" for personalized shoes in Germany close to the customer, using Cyber-physical production systems (CPPS).

# Factory as Internet of Machines



Main promise: Flexibility, Ease of reconfiguration, Robustness, Adaptability

# Реконфигурируемые фабрики

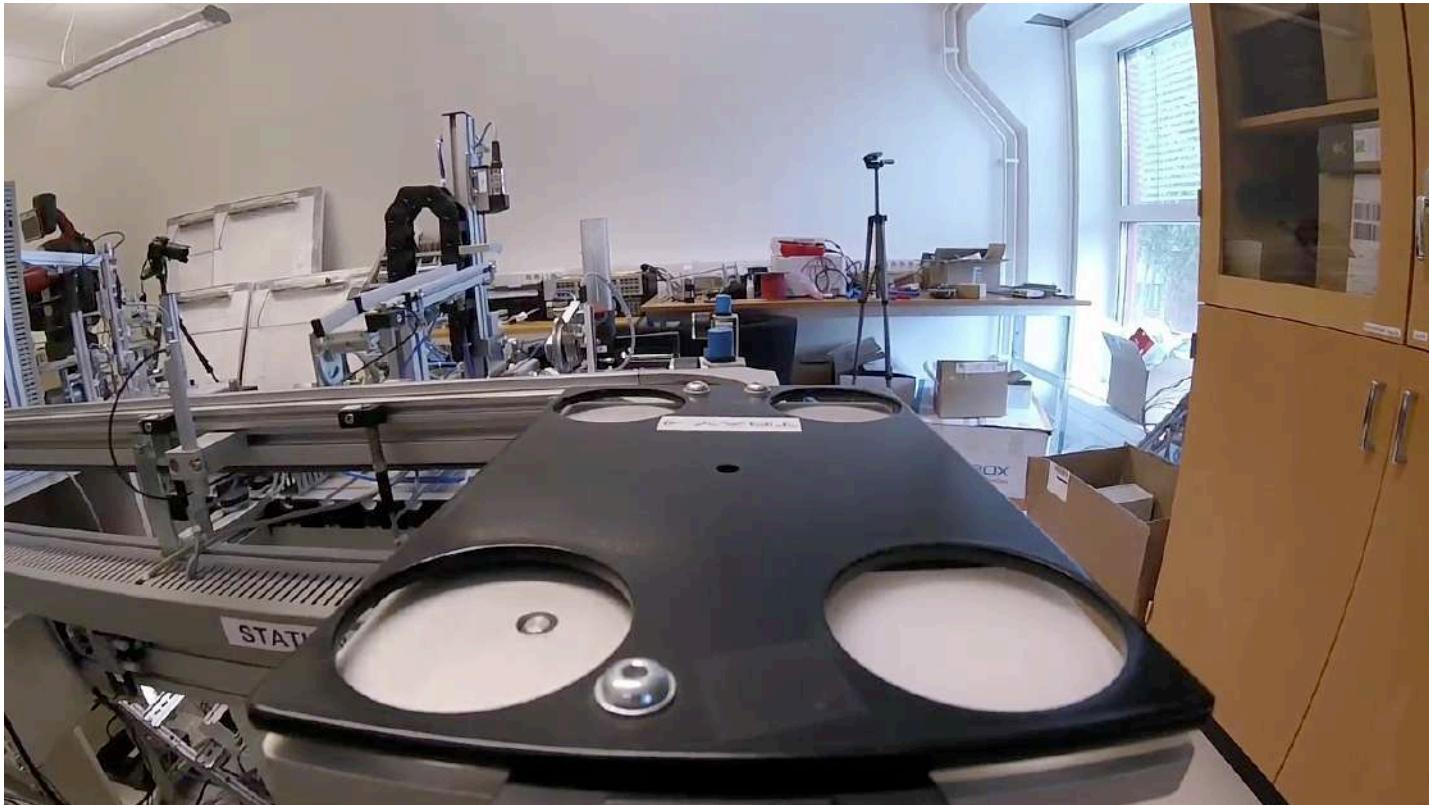




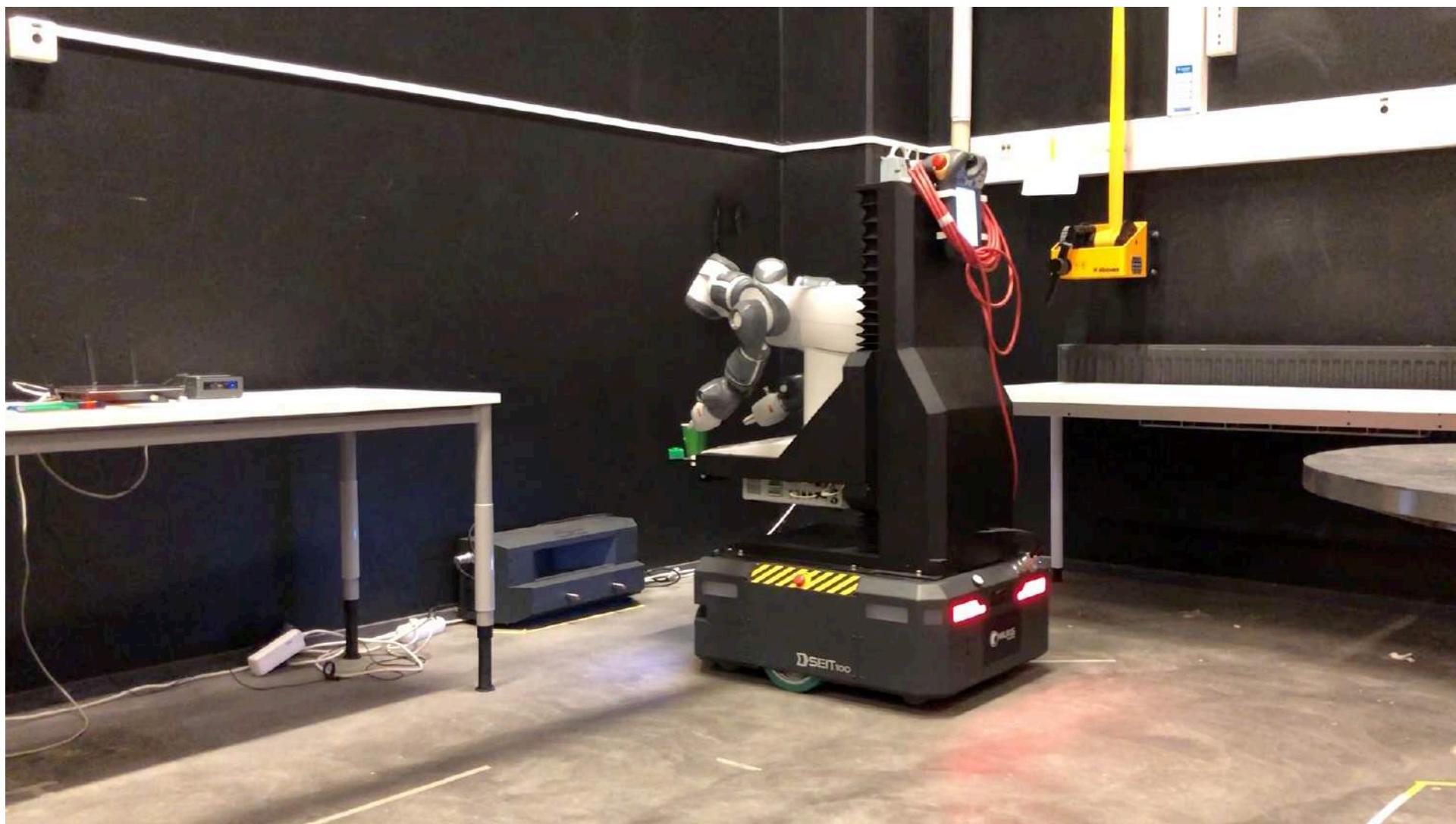
# AIC<sup>3</sup> laboratory, Luleå

[www.ltu.se/AICcube](http://www.ltu.se/AICcube)

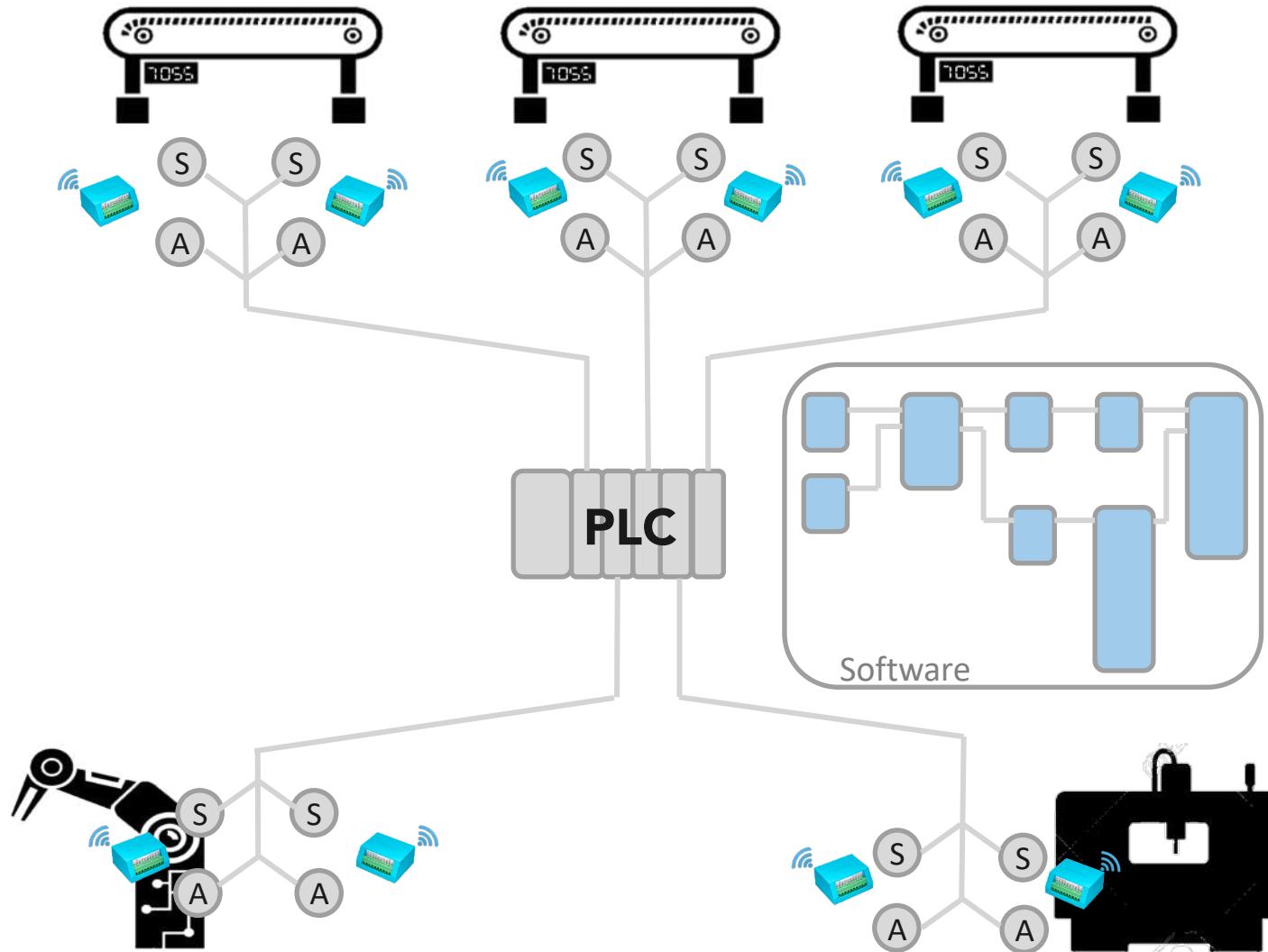
Automation, Industrial Computing, Communication and Control



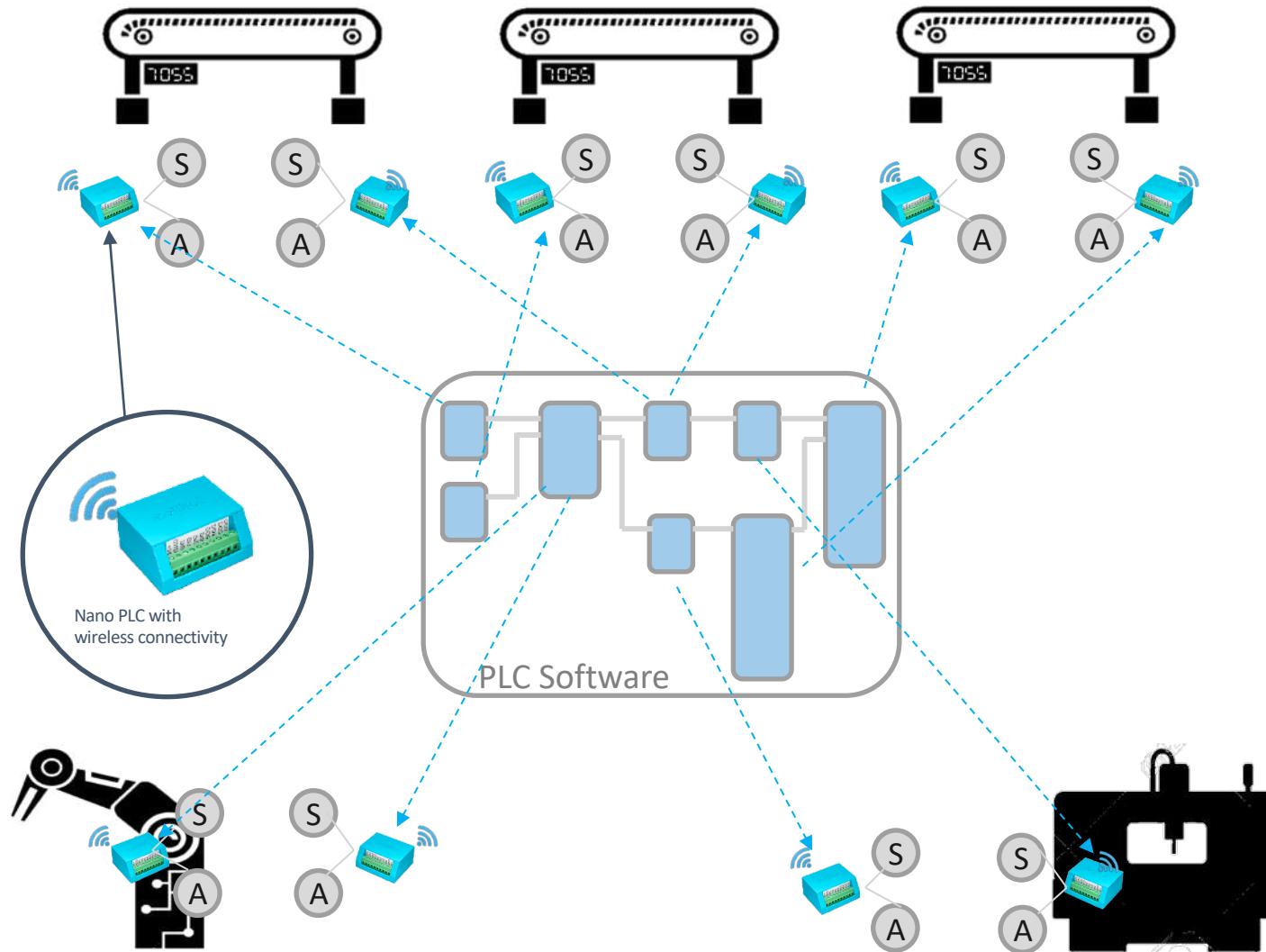
# Aalto Factory of the Future



# Автоматизация: смена архитектуры

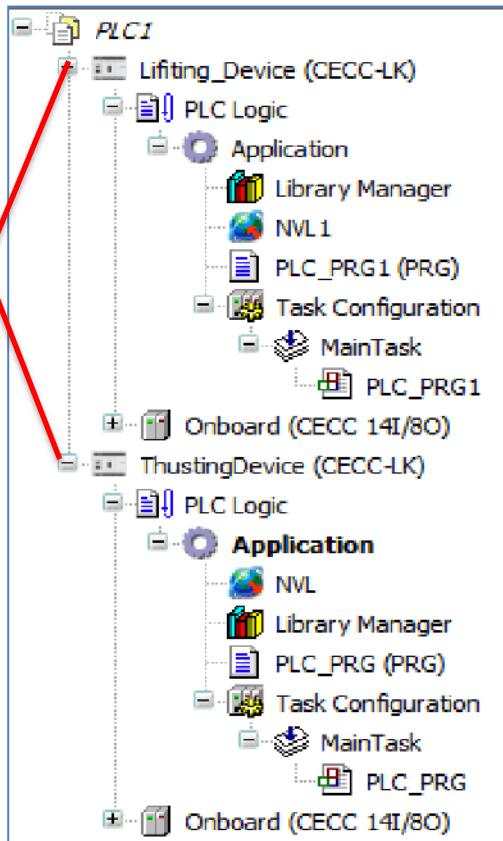
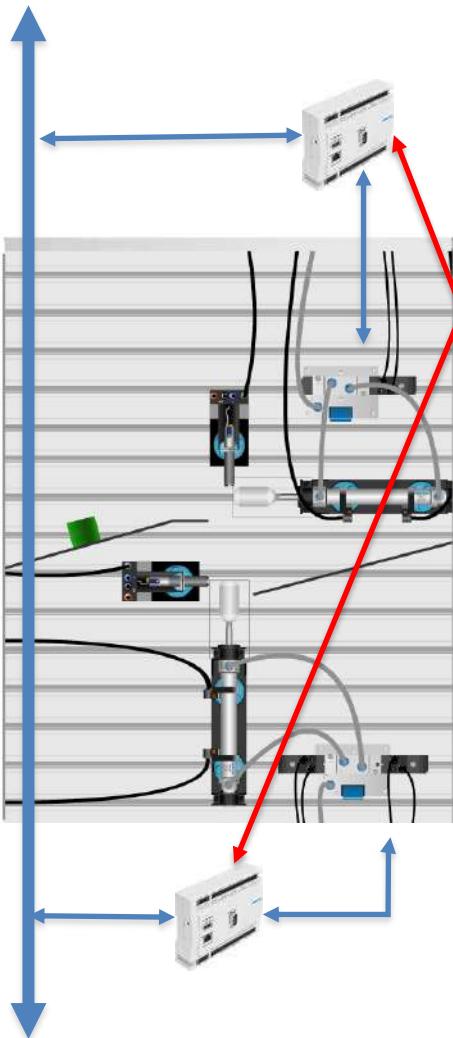


# Автоматизация: смена архитектуры



# Проблемы с моделью вычислений

Ethernet



```
PROGRAM PLC_PRG1
VAR
    LCExtended : BOOL;
    LCRetracted: BOOL;
    LuggageArrived: BOOL;
    ExtendLC: BOOL; → Actuator
    RetractLC: BOOL;
END_VAR
IF LuggageArrived AND NOT LCExtended THEN
    ExtendLC := TRUE;
    RetractLC := FALSE;
ELSIF LCExtended AND SharedVariable THEN
    ExtendLC := FALSE;
    RetractLC := TRUE;
END_IF
```

```
PROGRAM PLC_PRG
VAR
    TCRetracted: BOOL;
    TCExtended: BOOL;
    LuggageRaised: BOOL;
    ExtendTC: BOOL; → Actuator
    LuggageAway: BOOL;
END_VAR
IF LuggageRaised AND TCRetracted THEN
    ExtendTC:= TRUE;
    LuggageAway := FALSE;
    SharedVariable := FALSE;
ELSIF TCExtended AND NOT LuggageRaised THEN
    ExtendTC:= FALSE;
    SharedVariable := TRUE;
END_IF
```

# Using PLCs in a Distributed System: Working

Devices

- LiftingLuggage\_Sync
  - Device [connected] (CODESYS)
- PLC Logic
  - Application [run]
    - GVL
    - Library Manager
    - LiftCylProd (PRG)
    - LiftCylISBDv1 (PRG)
  - Model (PRG)
    - ThrustCylProd (PRG)
    - ThrustCylISBDv0 (PRG)
    - ThrustCylISBDv1 (PRG)
    - ThrustCylISBDv2 (PRG)
    - ThrustCylISBDv3 (PRG)
  - Task Configuration
    - Lifting\_Device
      - LiftCylProd
    - ModelTask
    - Model
  - Thrusting\_Device
    - ThrustCylProd
  - VisualizationManager
  - Visualization

Thrusting\_Device LiftCylProd

Device.Application.LiftCylProd

```
1 IF LuggageArrived[FALSE] AND NOT LCExtended[TRUE] THEN
2   ExtendLC[TRUE] := TRUE;
3   RetractLC[FALSE] := FALSE;
4 ELSIF LCExtended[TRUE] AND SharedVar[FALSE] THEN
5   ExtendLC[TRUE] := FALSE;
6   RetractLC[FALSE] := TRUE;
7 END_IF[RETURN]
```

Visualization

For testing purposes only:

- Extend lifting cylinder
- Retract lifting cylinder
- Extend thrust cylinder

Lifting\_Device

Configuration

Priority ( 0..31 ): 10

Type: Cyclic Interval (e.g. t#200ms): t#100ms

Watchdog

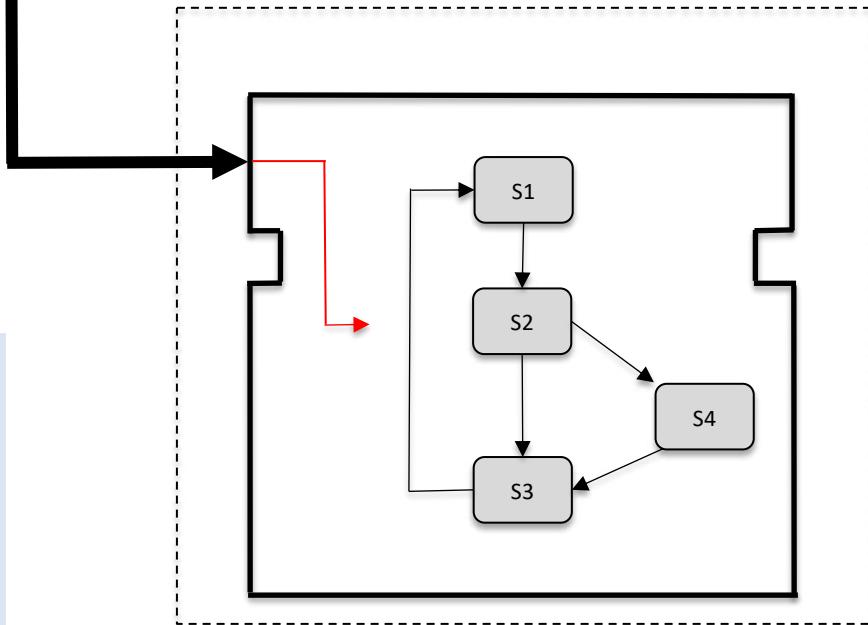
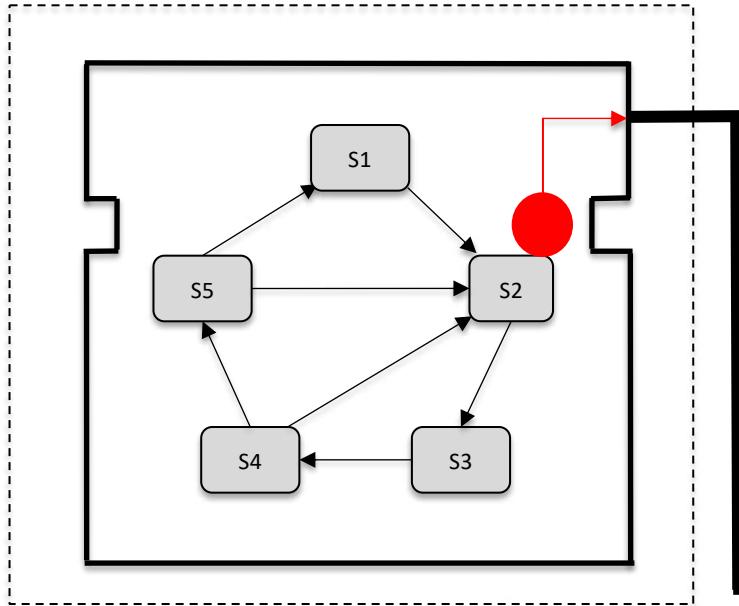
Enable

Time (e.g. t#200ms): ms

Sensitivity: 1

Messages - Total 4 error(s), 0 warning(s), 4 message(s) 0

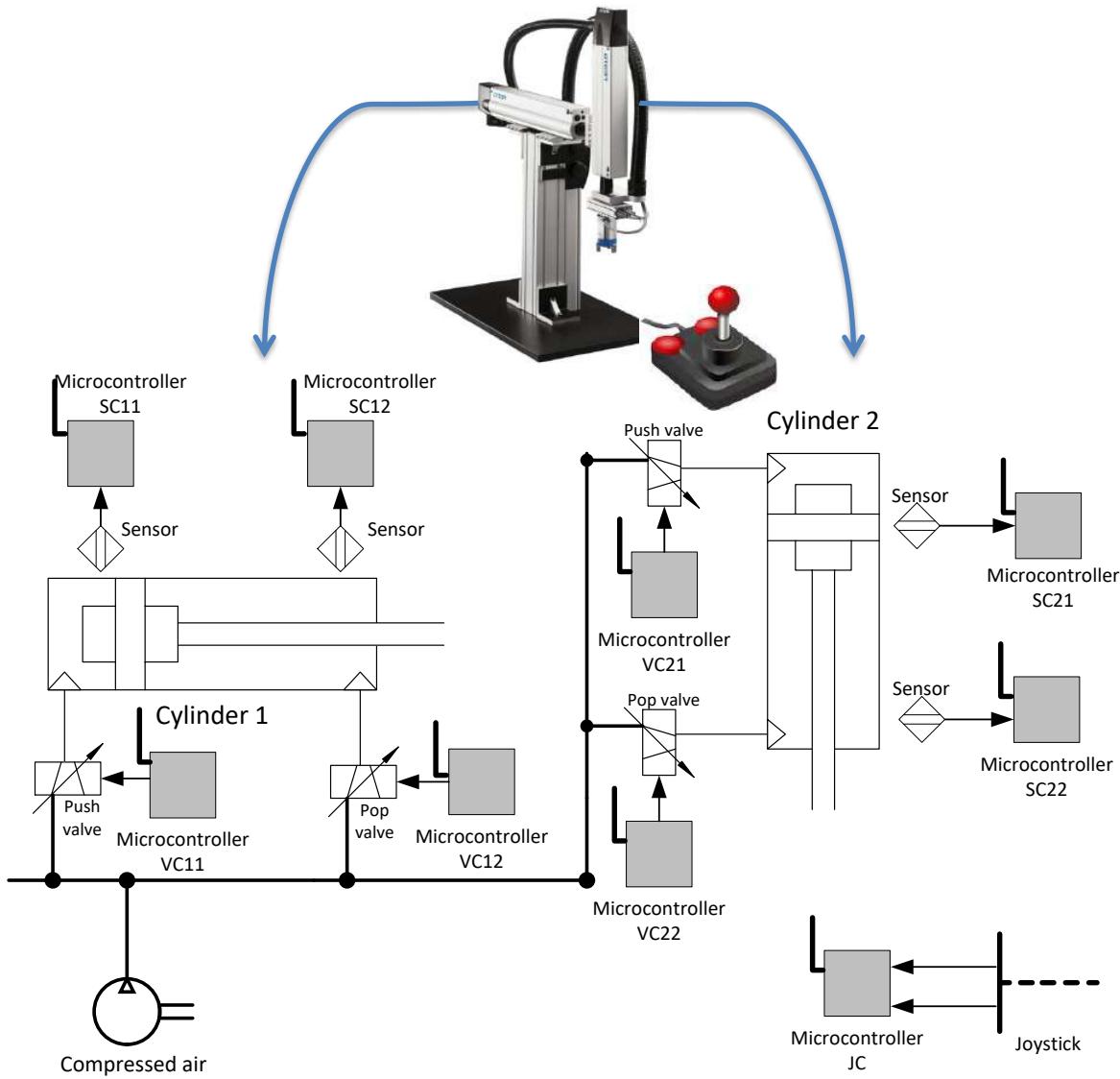
# Модель вычислений основанная на событиях



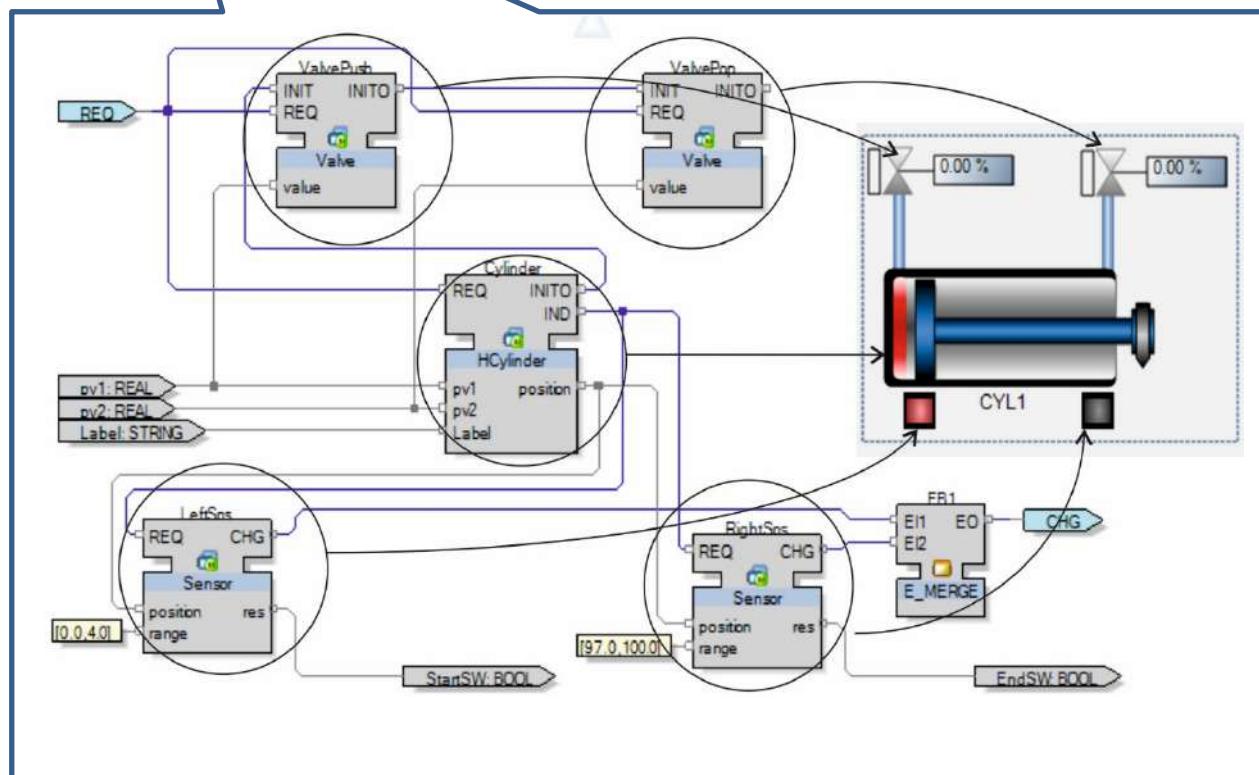
Свойства:

- Детерминированность в распределённой системе – не зависит от циклов обновления данных

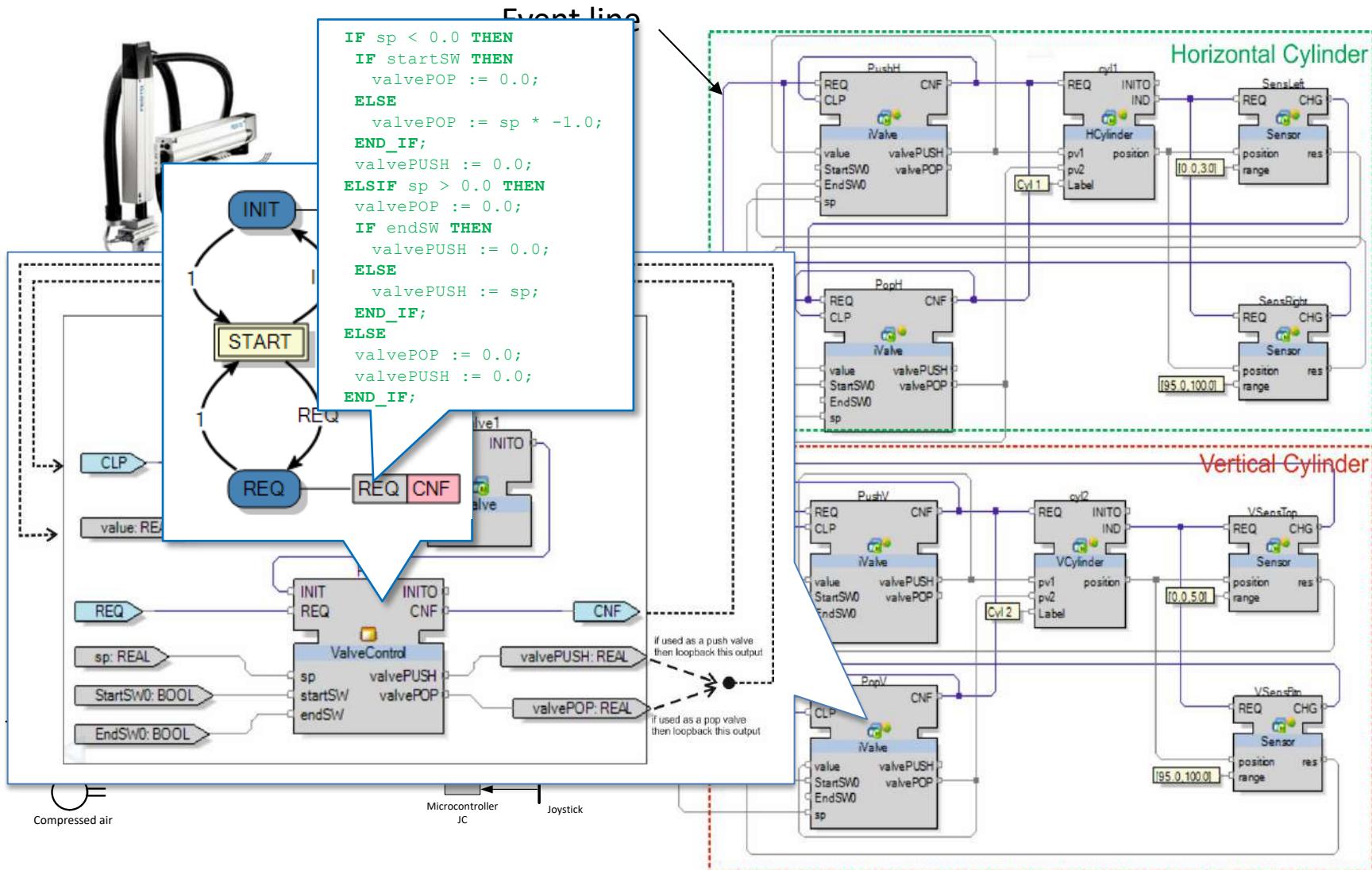
# Mechatronic IoT



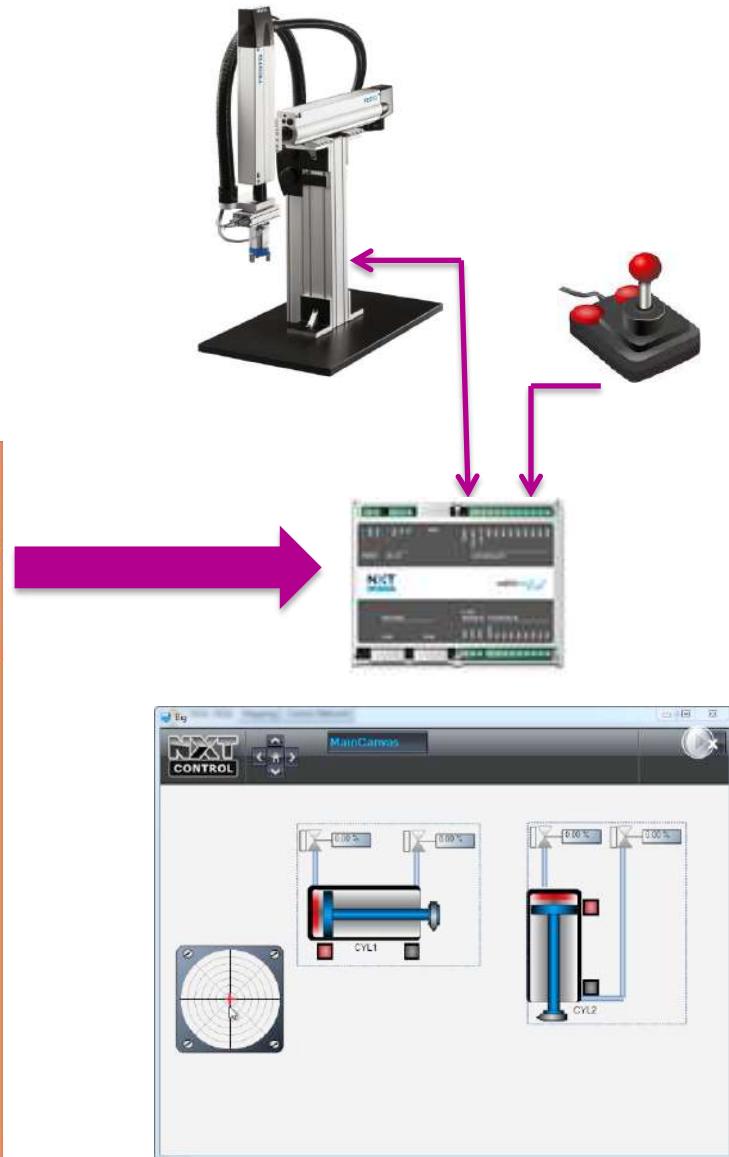
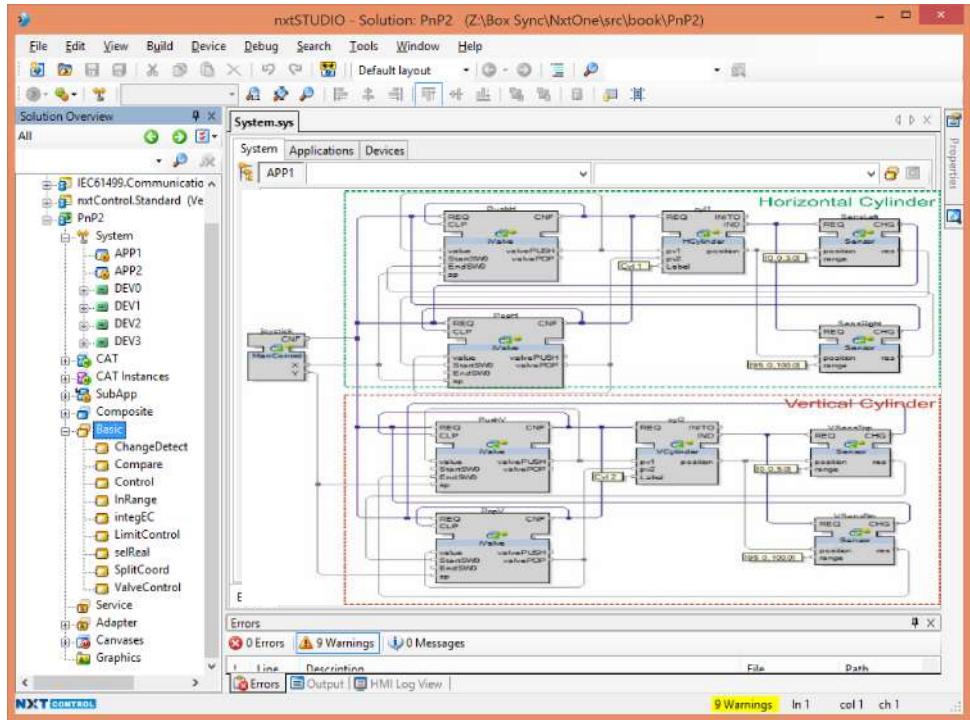
# Model of a Cylinder



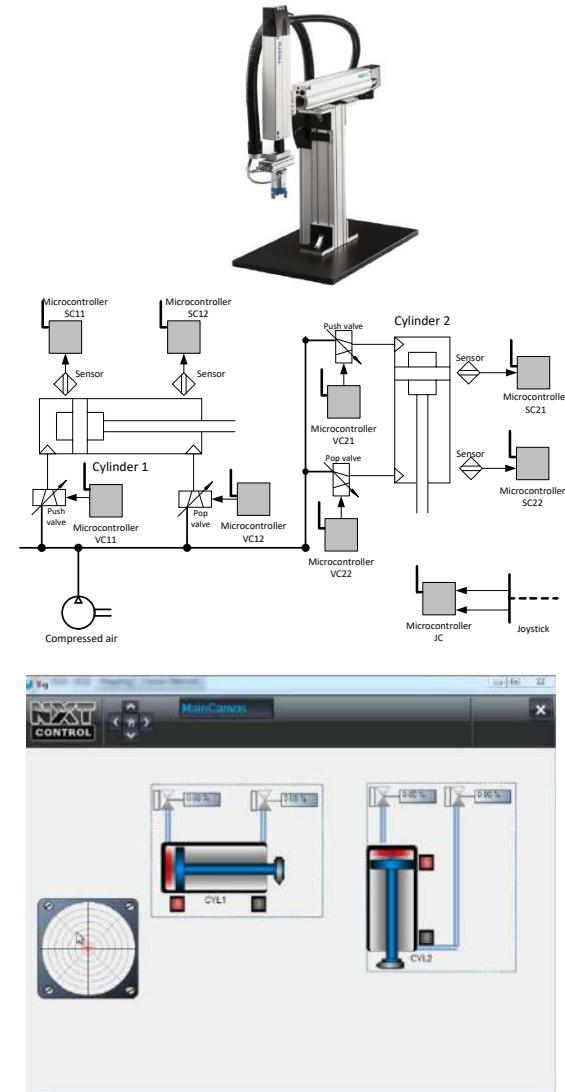
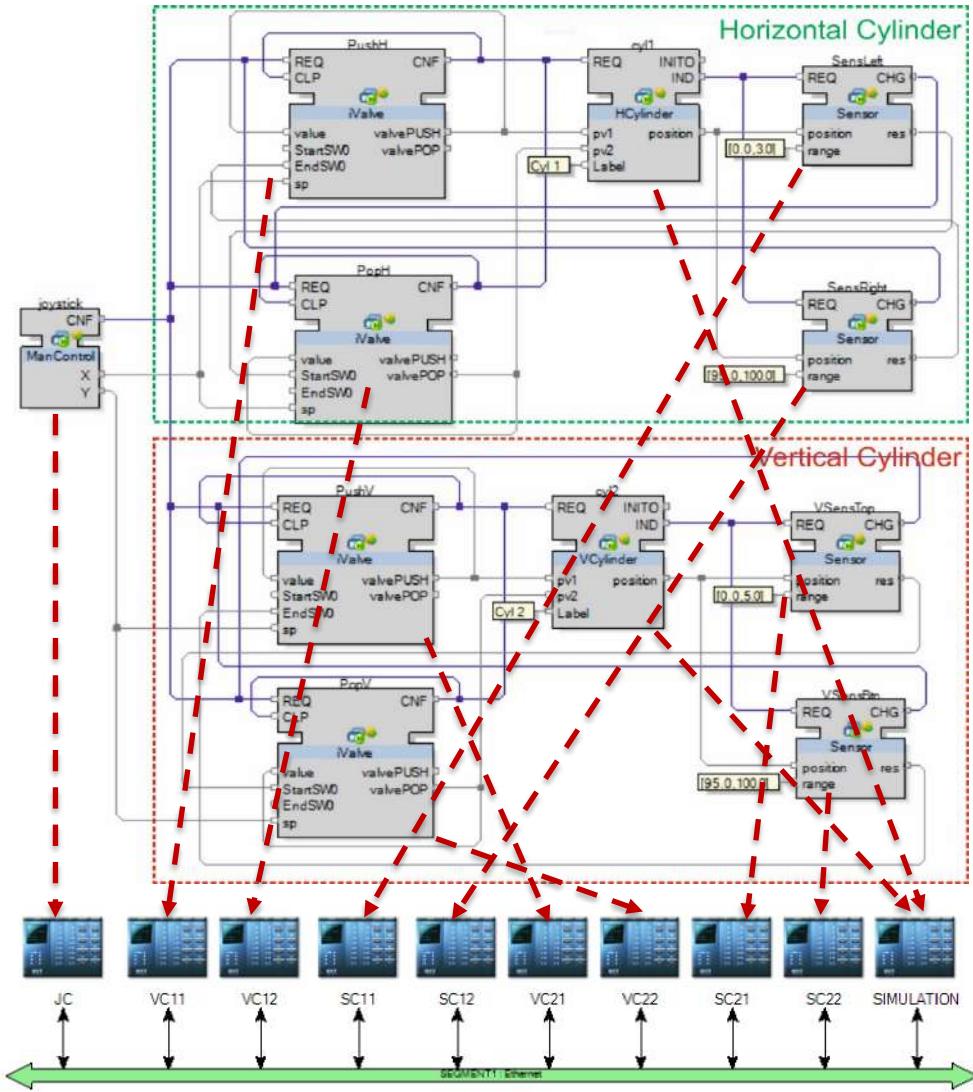
# Control software of the manipulator



# Deployment to a single control node



# Distributed automation system

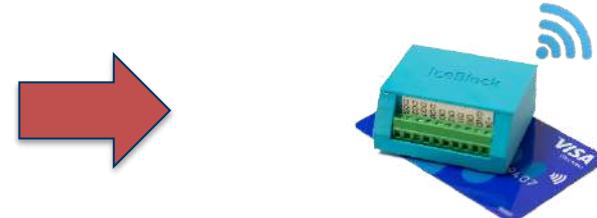


# Future networking PLCs

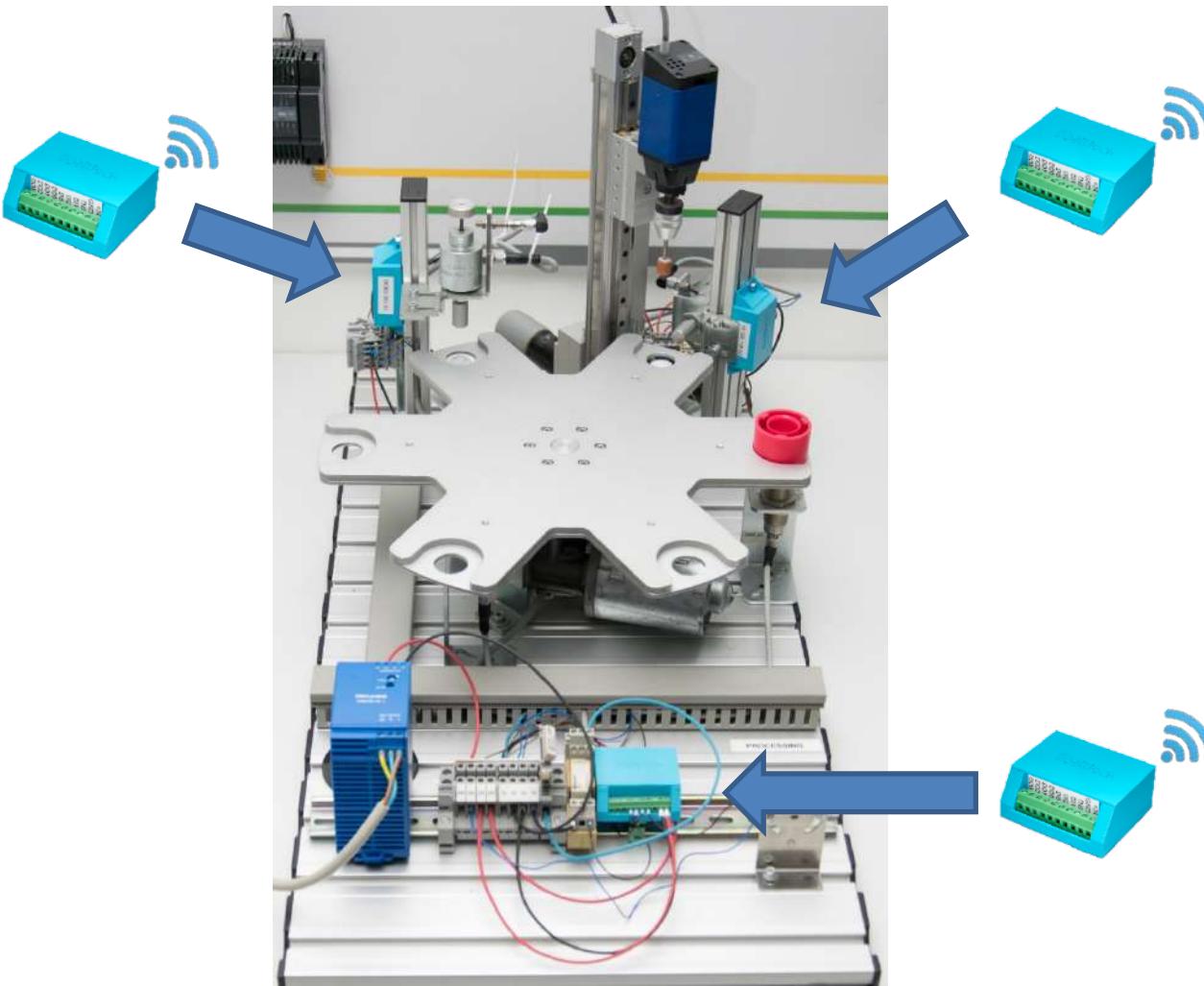
Current PLC



Future Wireless PLC



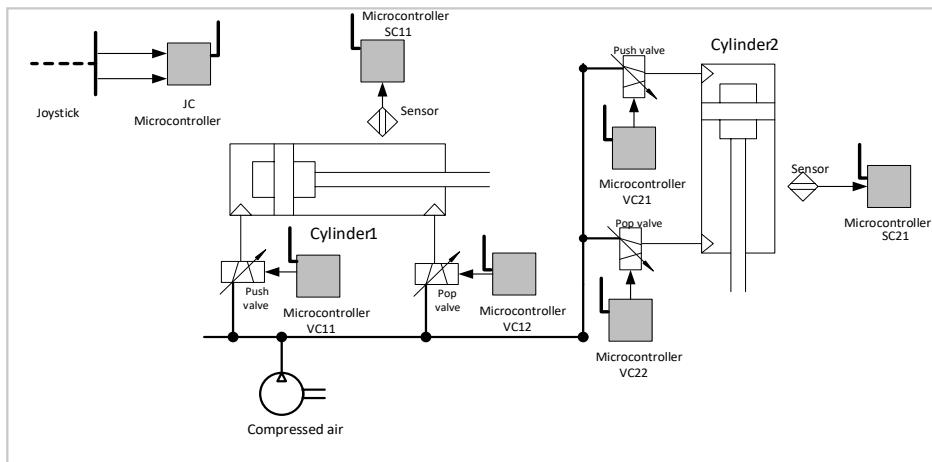
# Prototype of intelligent mechatronic system with wireless communication



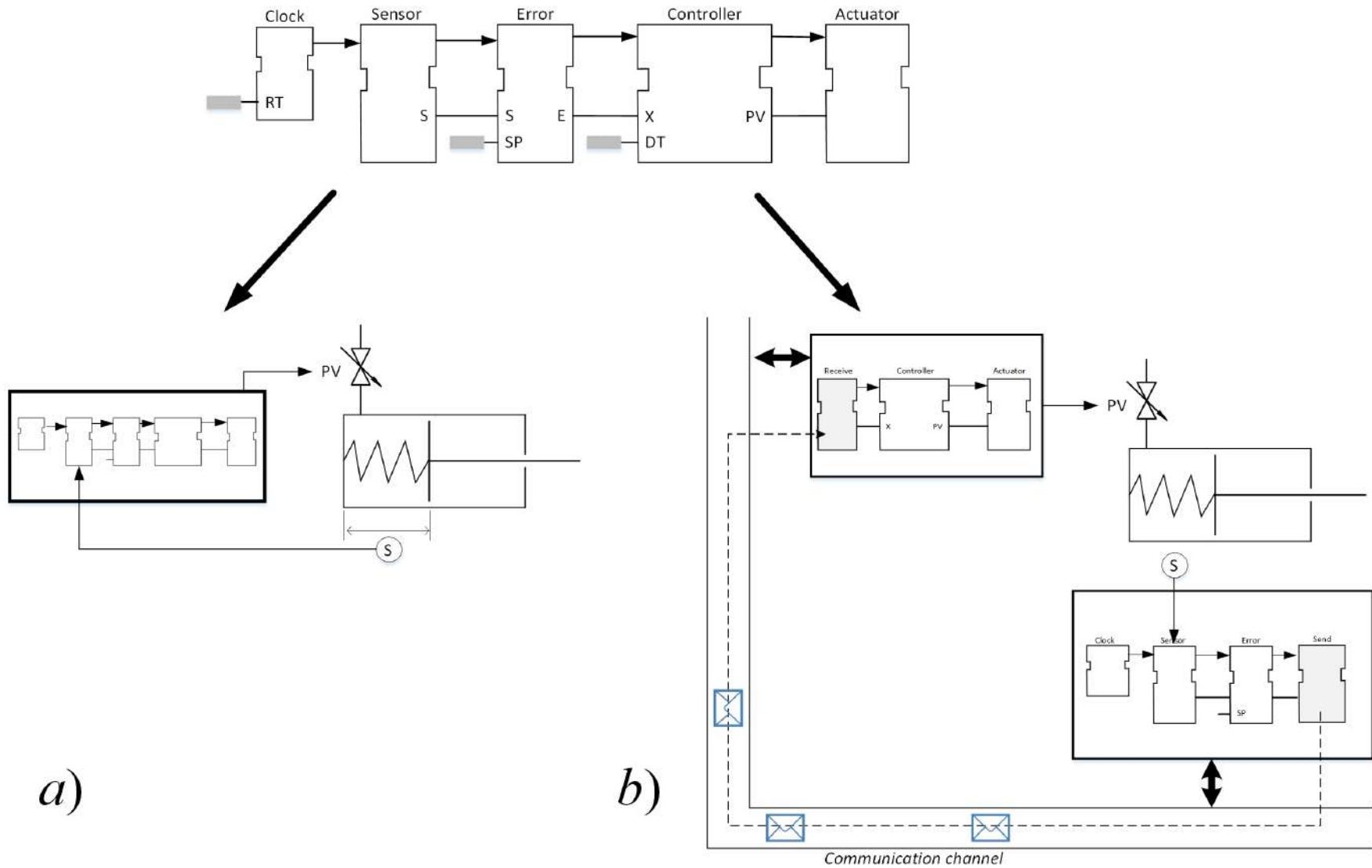
# Aalto Factory of the Future at Scanautomatic Fair 2018



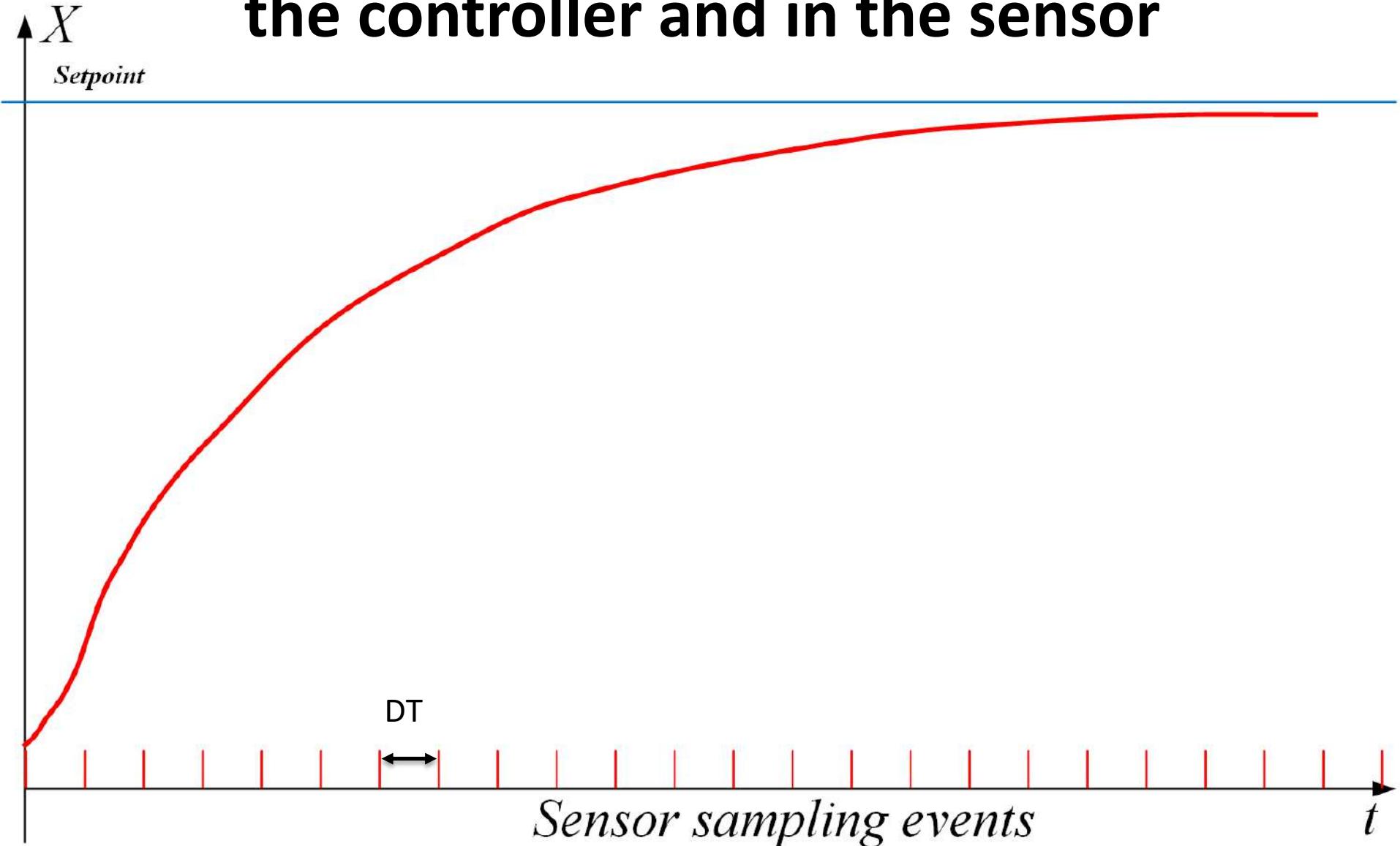
# Как сделать вычисления надёжными?



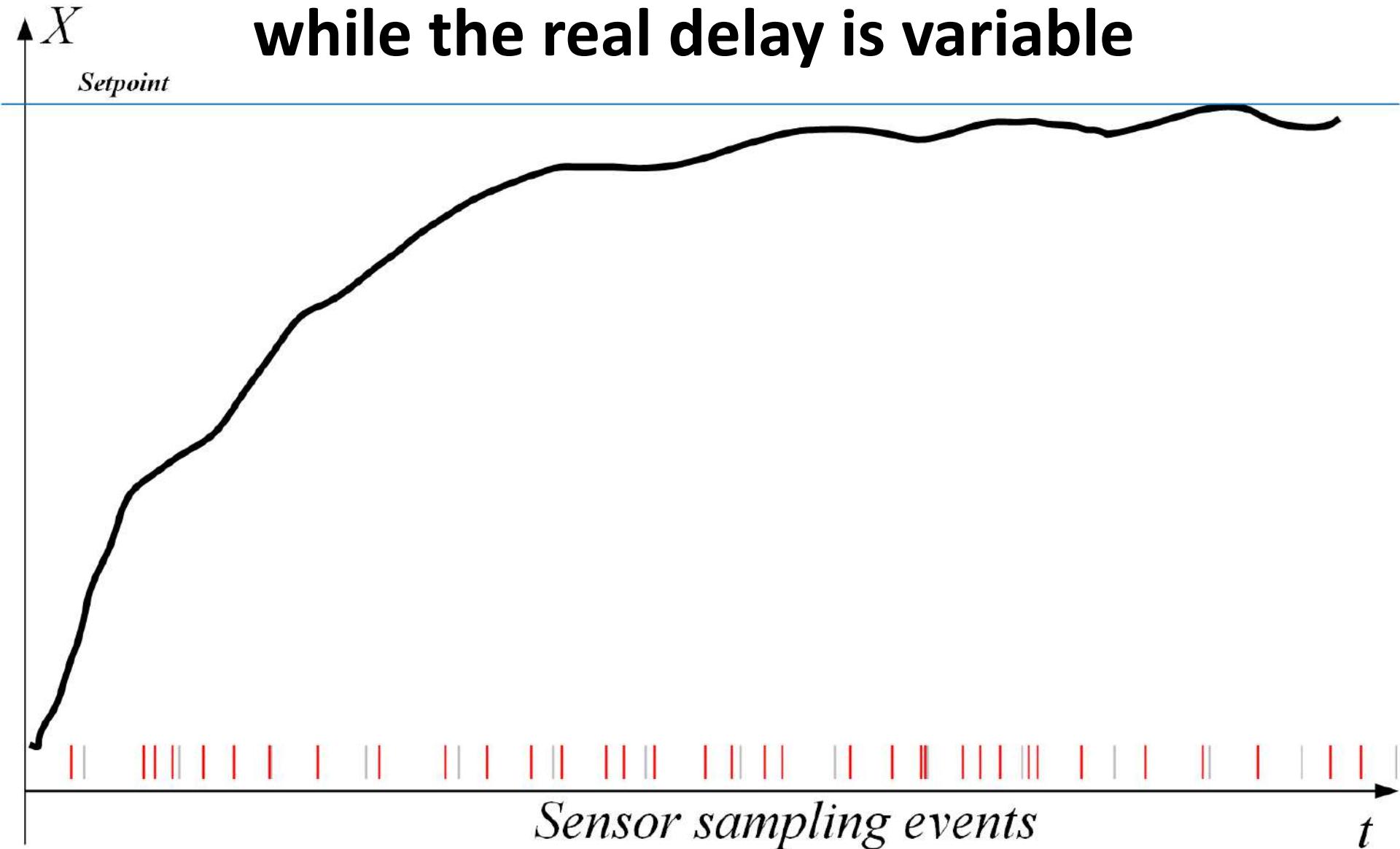
# Central control vs. Decentralised control



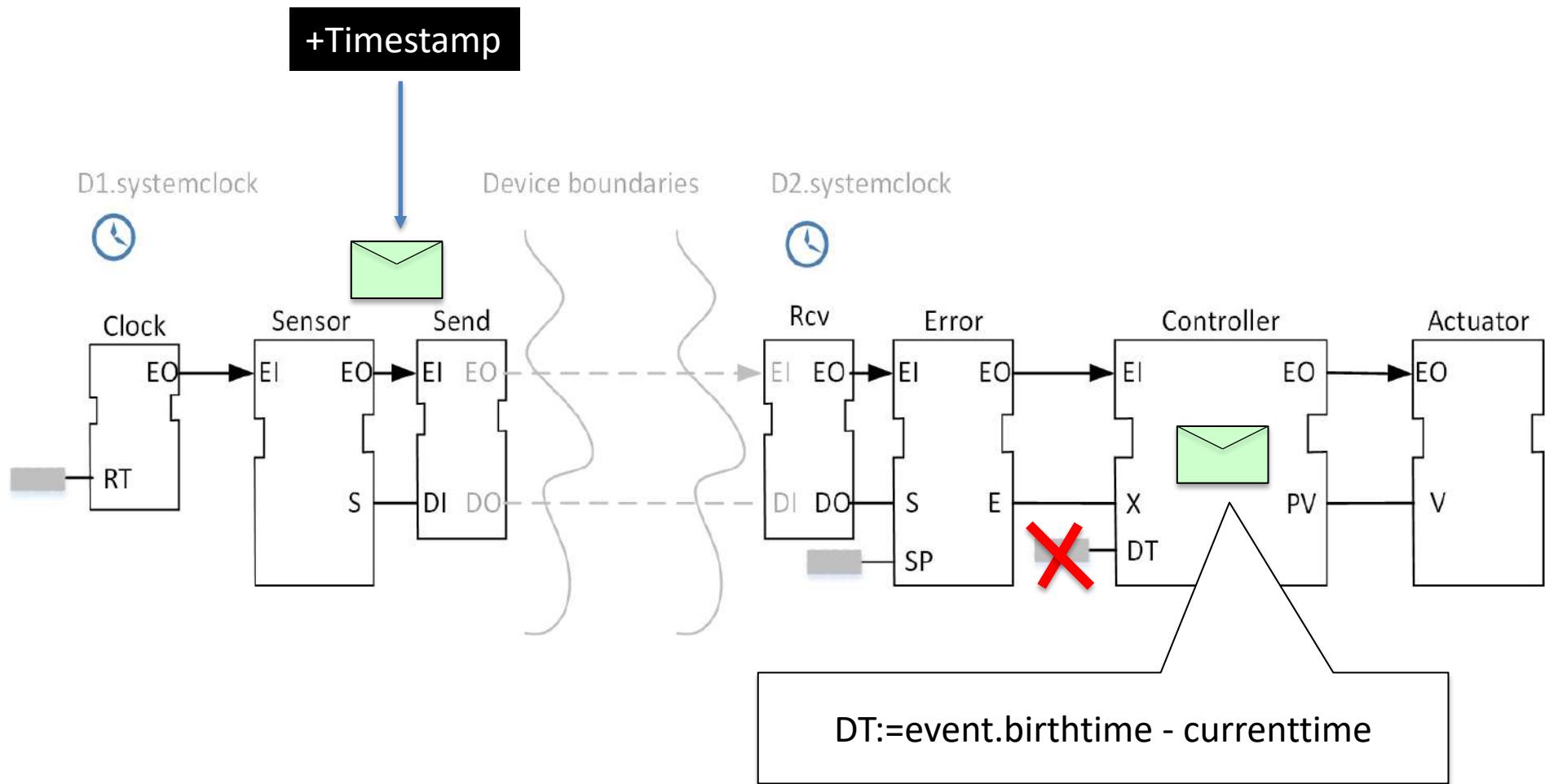
# "Ideal" case: sampling delay DT is constant in the controller and in the sensor



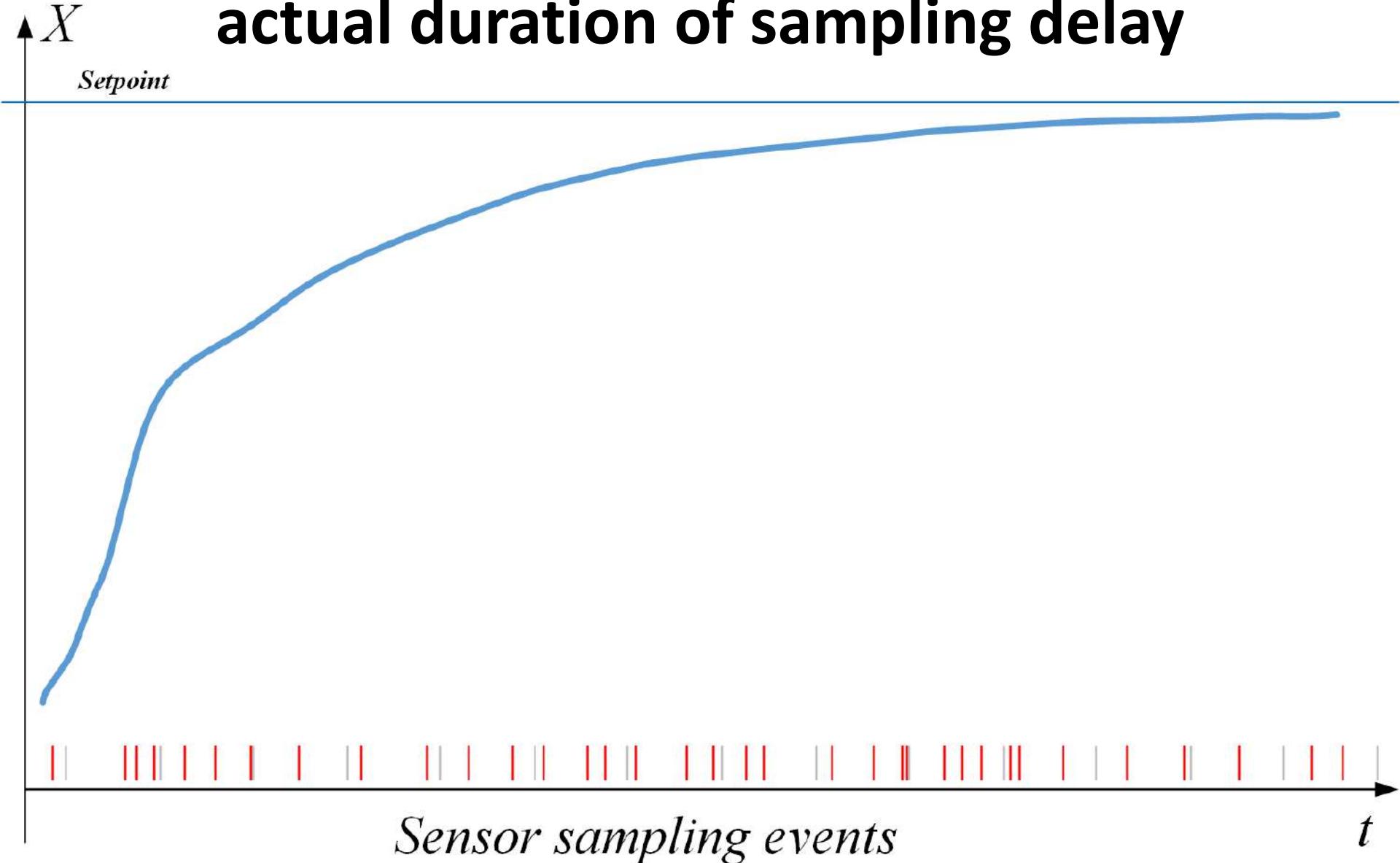
# Controller uses constant sampling delay, while the real delay is variable



# Timestamping of events to know the actual sampling interval



# Using event timestamping to determine actual duration of sampling delay



# Новые вызовы: как сохранить производительность?

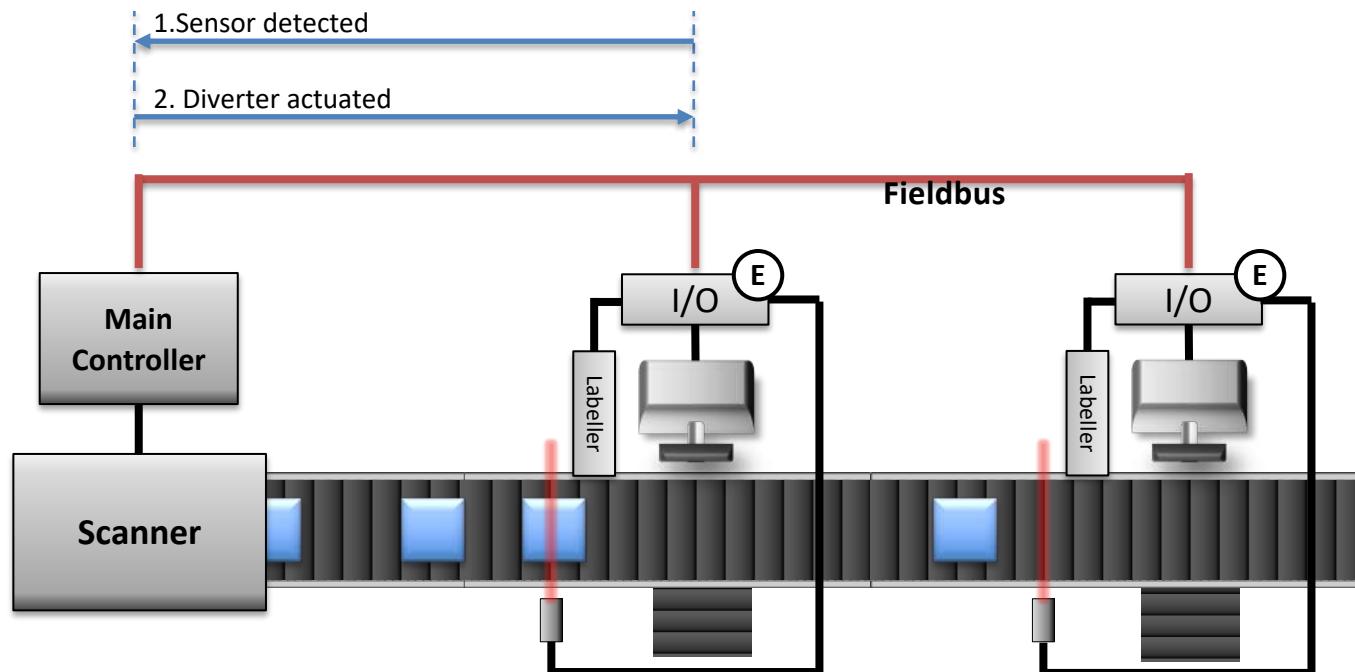


# Изменение парадигмы управления



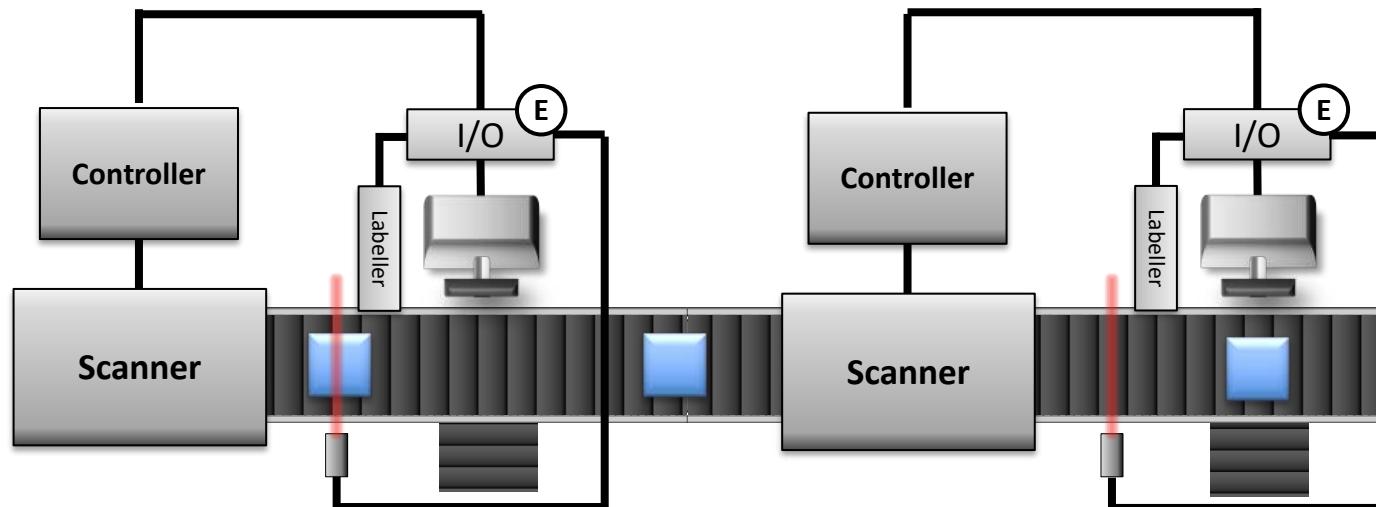
# Традиционная централизованная архитектура

Проблема: скорость конвейера ограничена скоростью передачи информации от датчика к контроллеру и от контроллера к исполнительному механизму



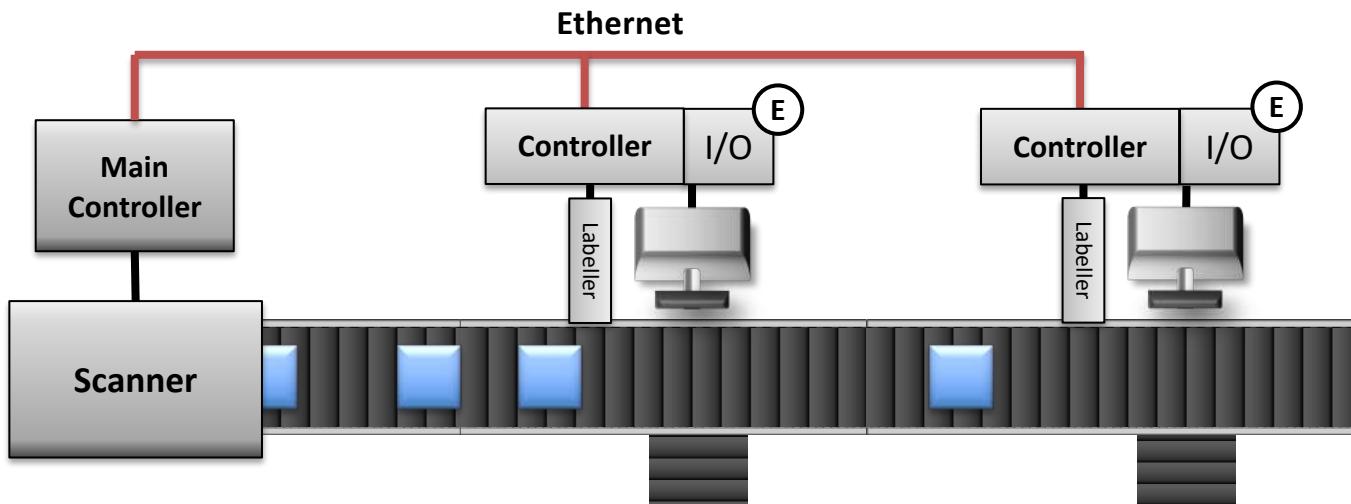
# Идеальная альтернатива: полностью распределённая архитектура

- Control algorithm fully distributed
- Dedicated scanner at every divert point
- No communication required
- Ideal but unrealistic configuration

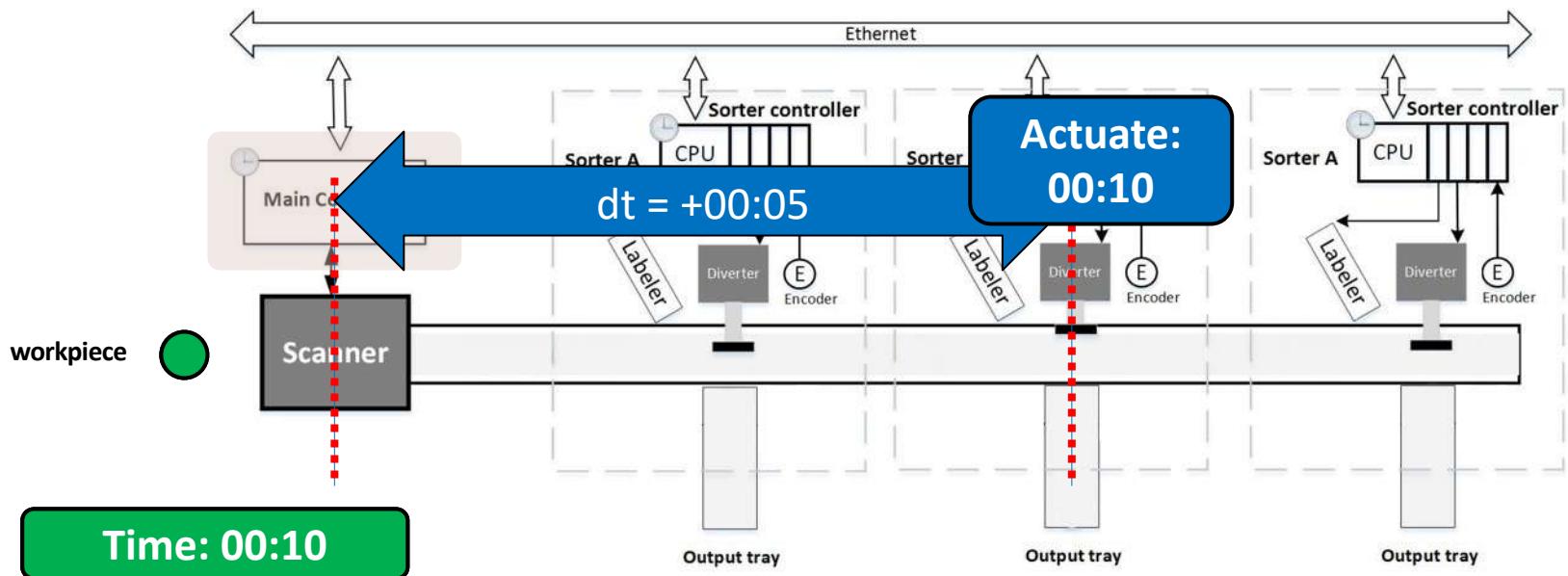


# Гибридная конфигурация

- Single main controller at scan point
- Lightweight controllers with I/O at divert points
- No sensor required

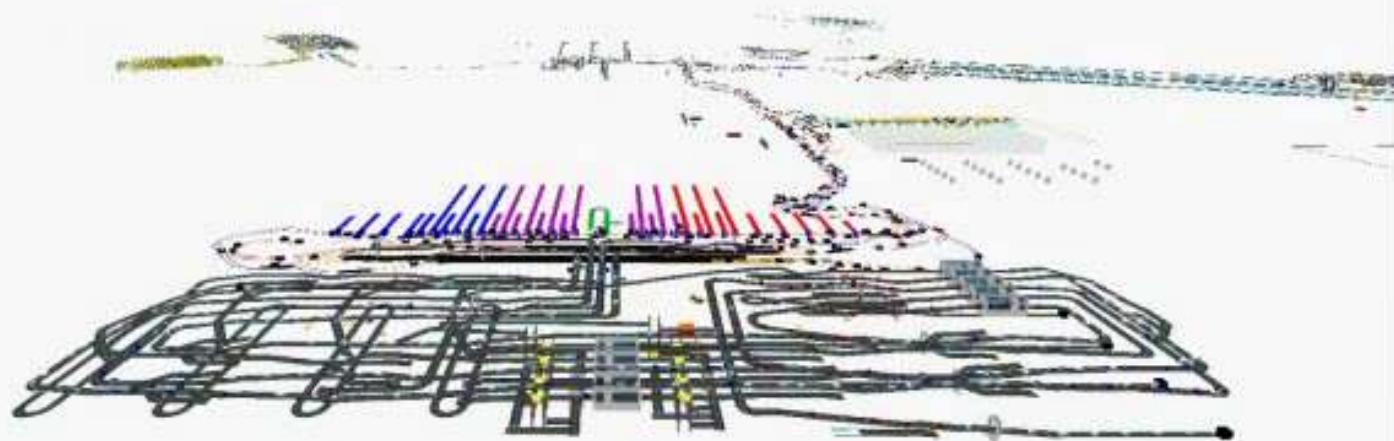


# Метод комбинированных событийно-временных вычислений

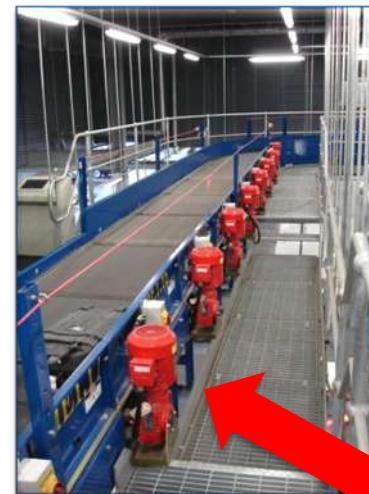
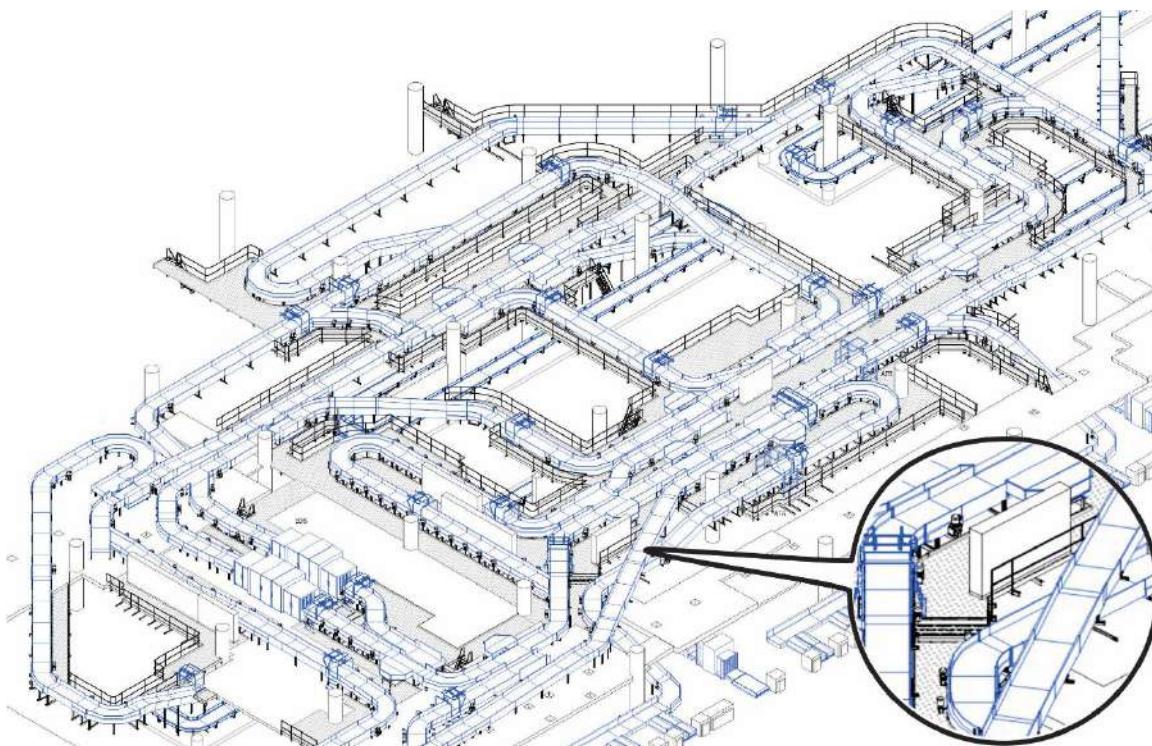


# Как справиться со сложностью?

## Пример: Airport Baggage Handling Systems



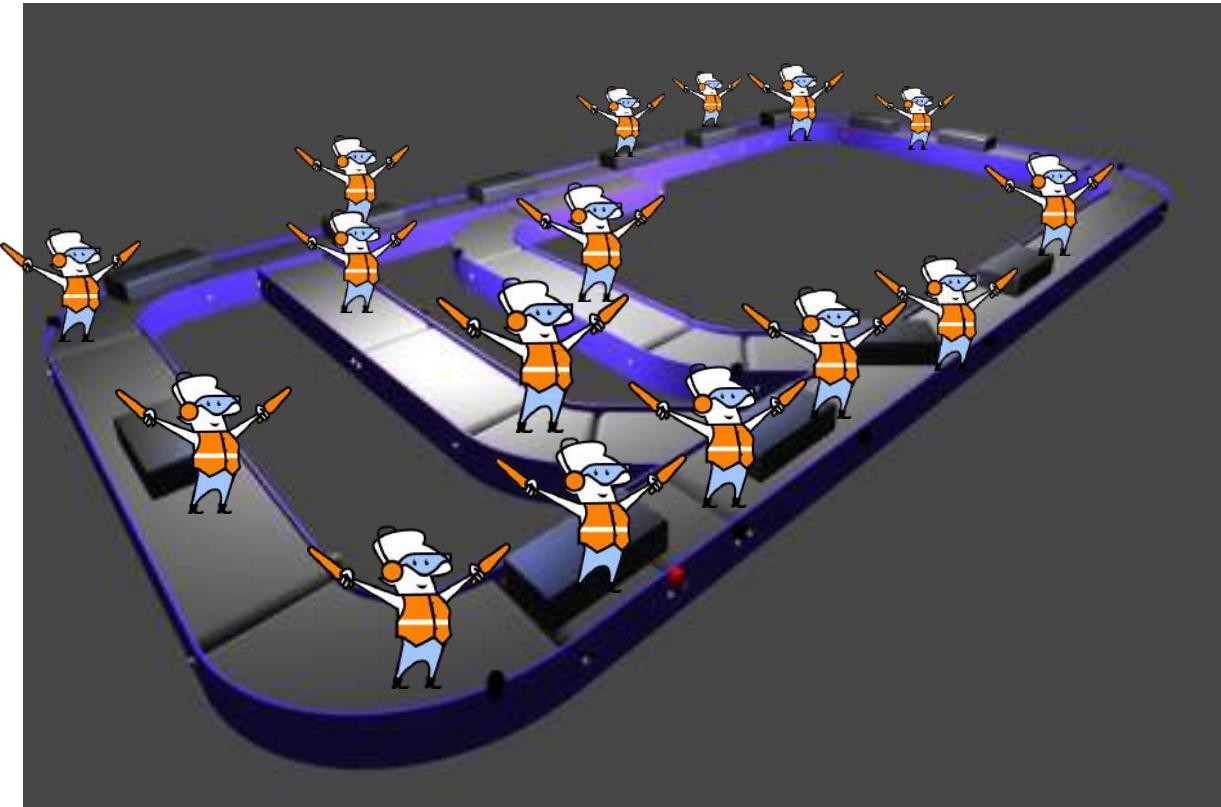
# Airport Baggage Handling Systems



IoT ready

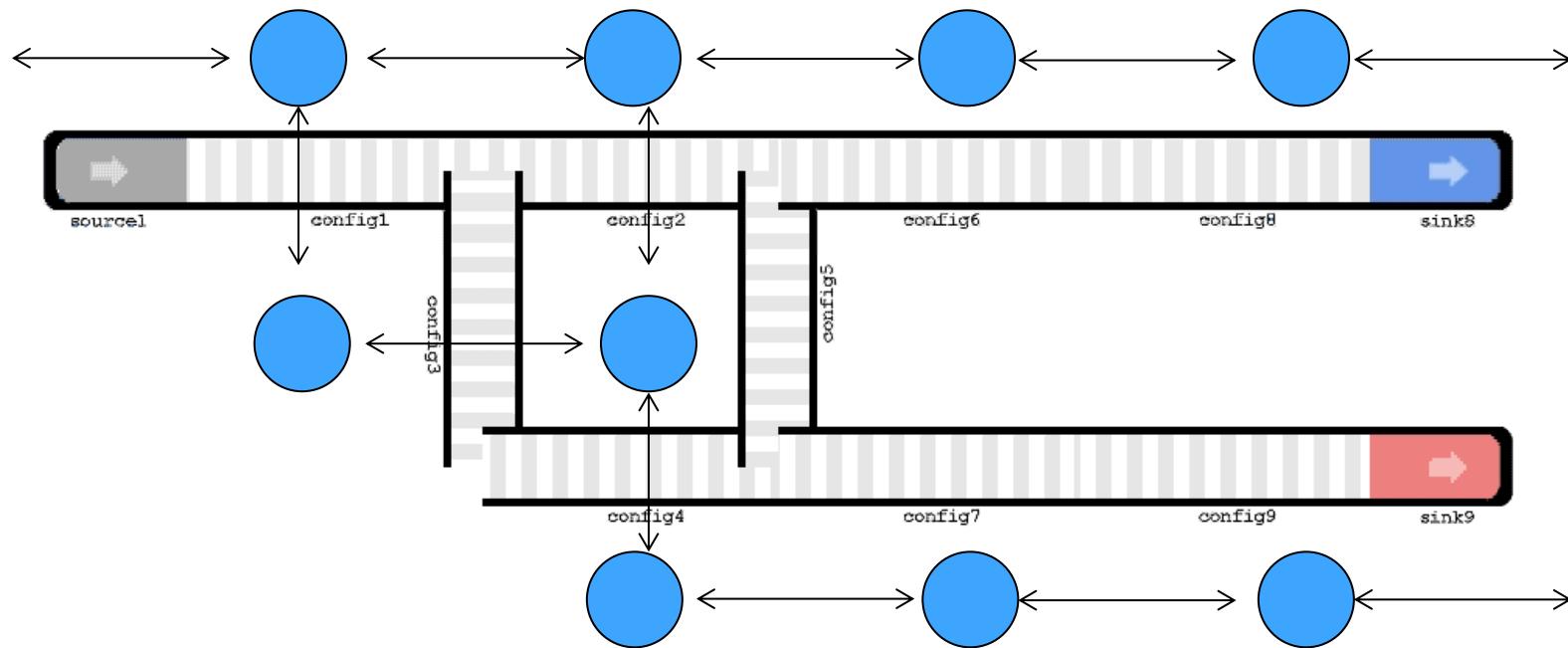


# Distributed Multi-agent Control

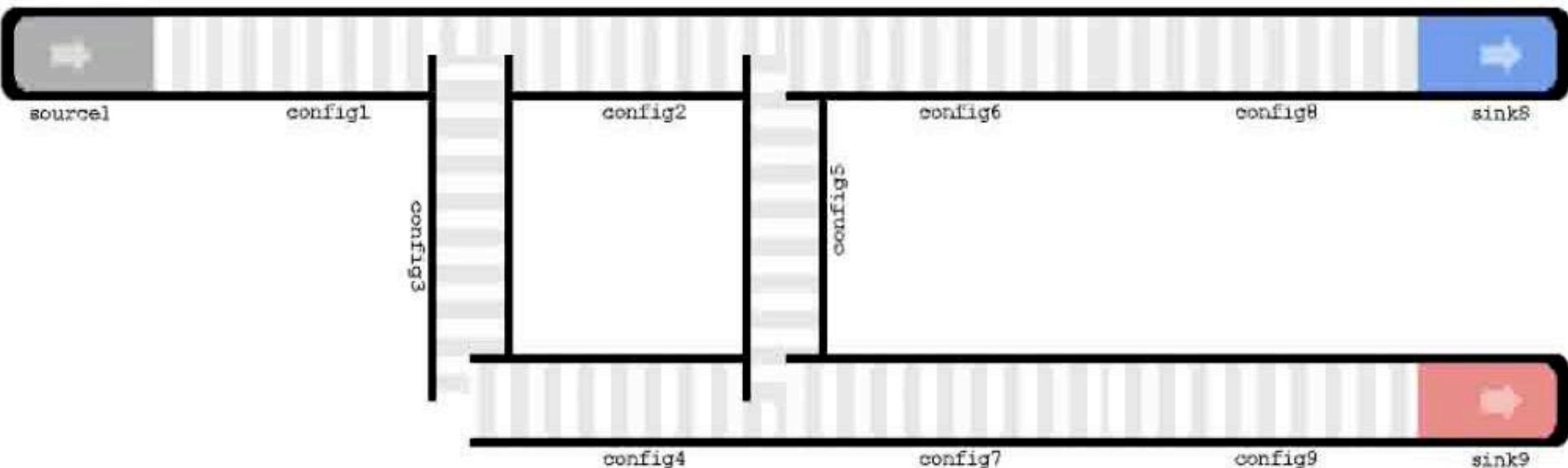
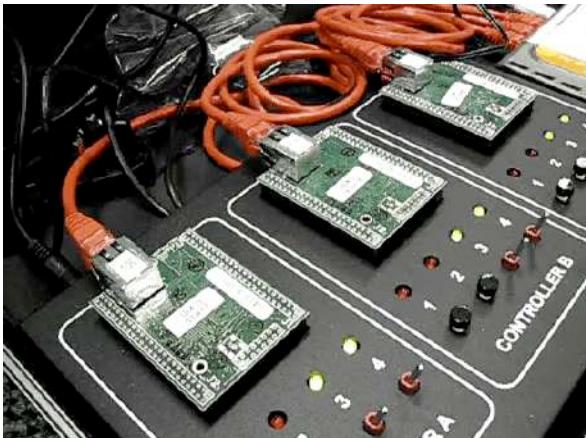


# Distributed Algorithms

Merging, Diverting, Fault tolerance handled using fully distributed collaboration of autonomous conveyor sections

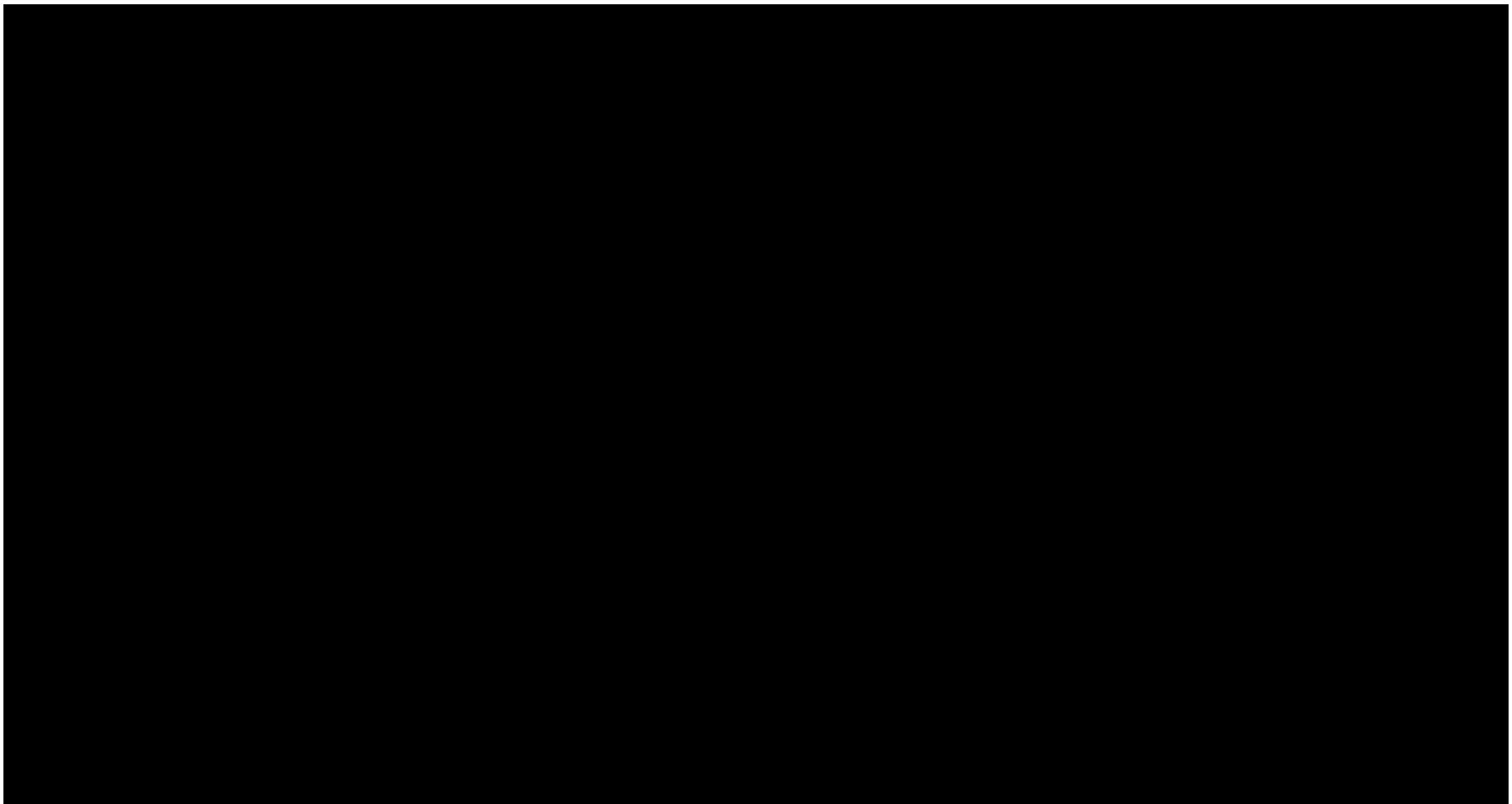


# Fault tolerance: re-routing on the fly

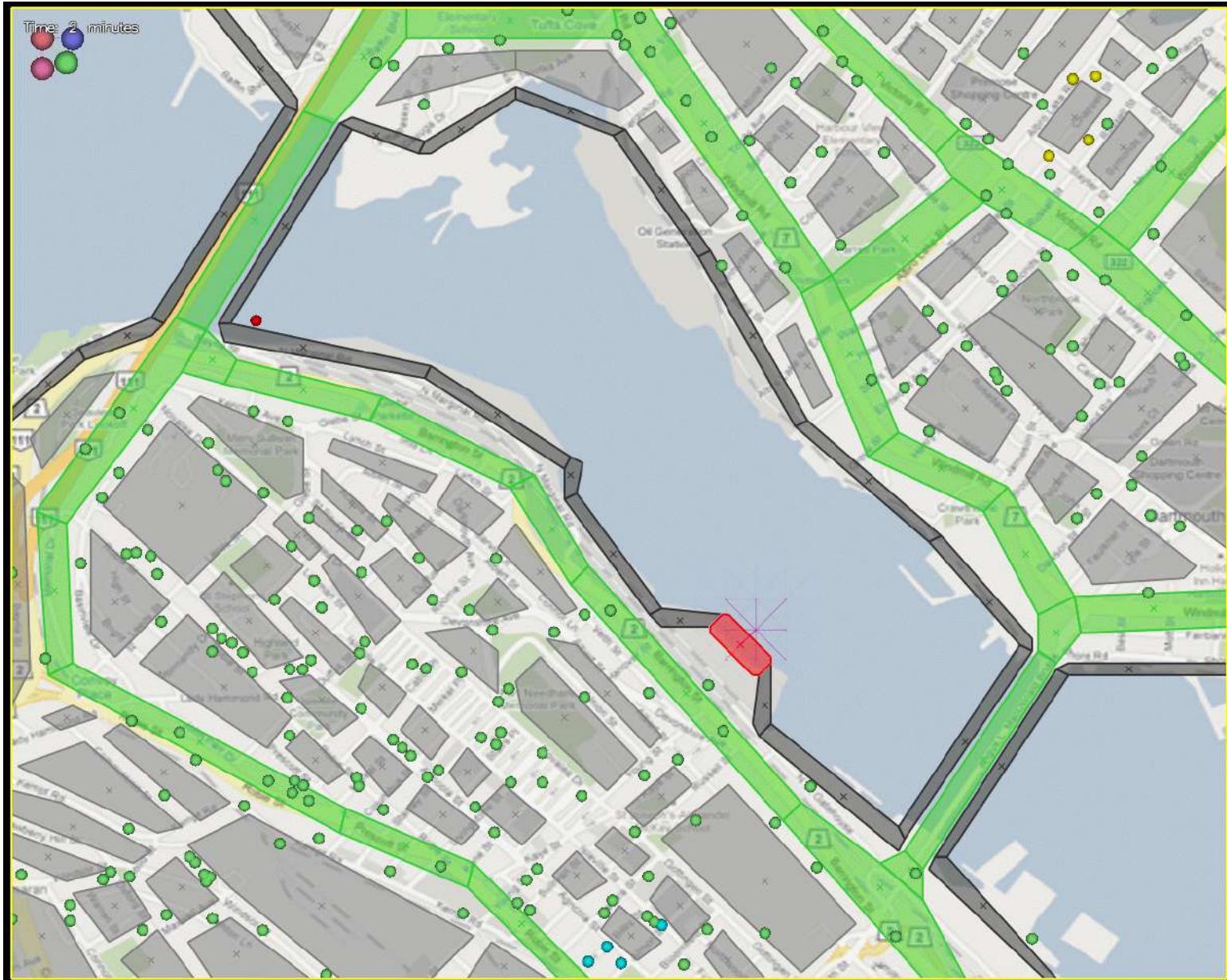


J. Yan, V. Vyatkin, “Distributed Software Architecture Enabling Peer-to-Peer Communicating Controllers”, *IEEE Transactions on Industrial Informatics*, 9(4), 2013, pp. 2200-2209

# Agents: self-organisation bio-inspiration

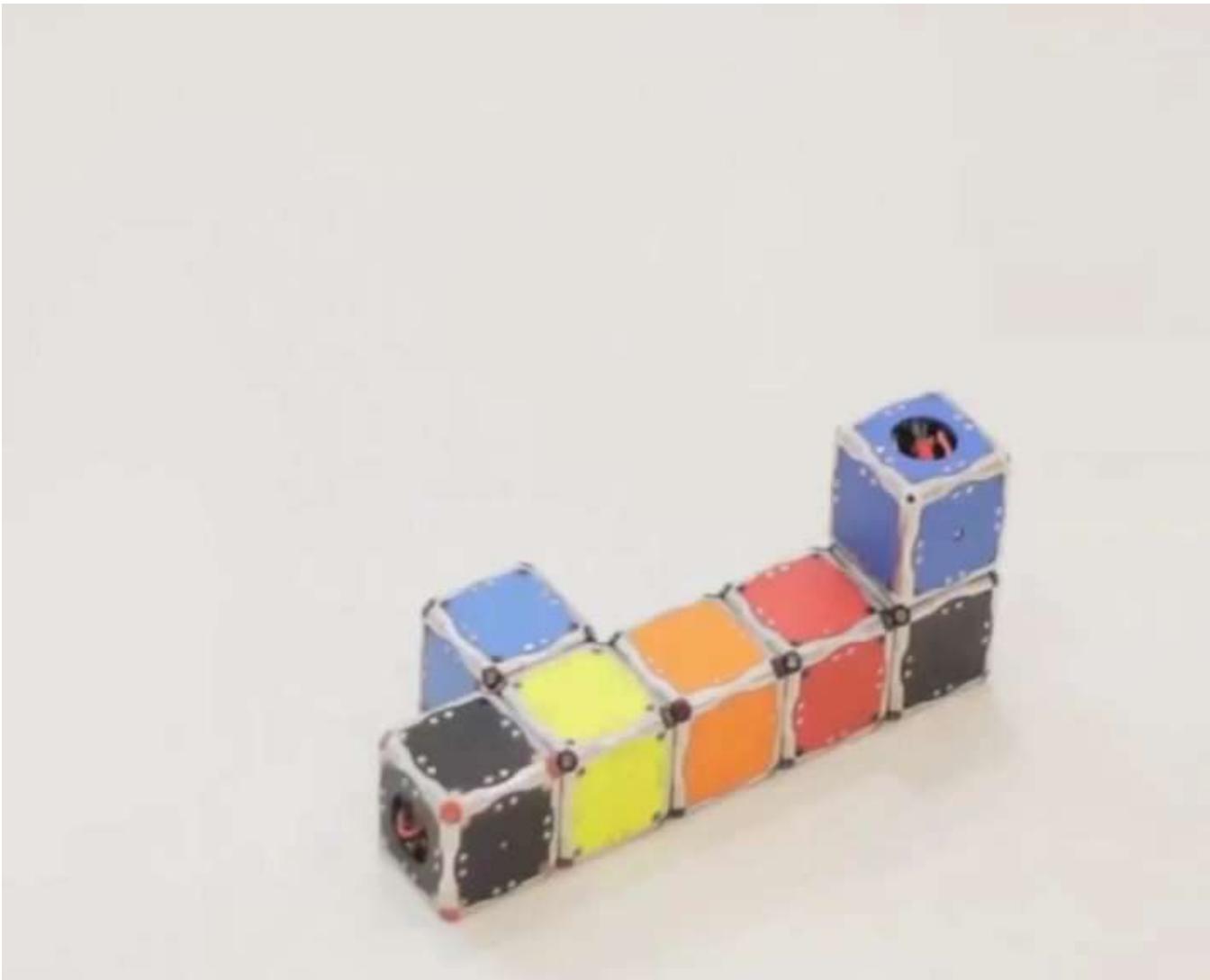


# Multi-Agent Simulation





# Self-configuration Dream





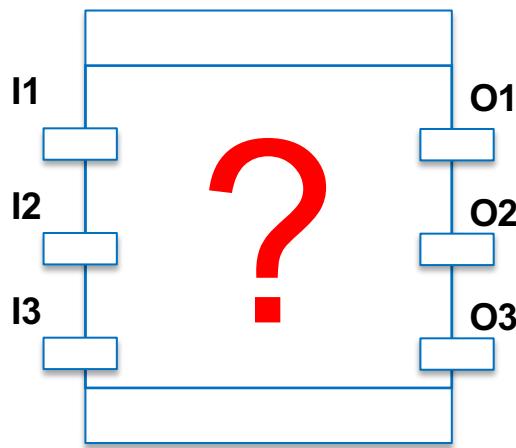
# Intelligent Automation Component

Pneumatic Cylinder with  
Embedded HW

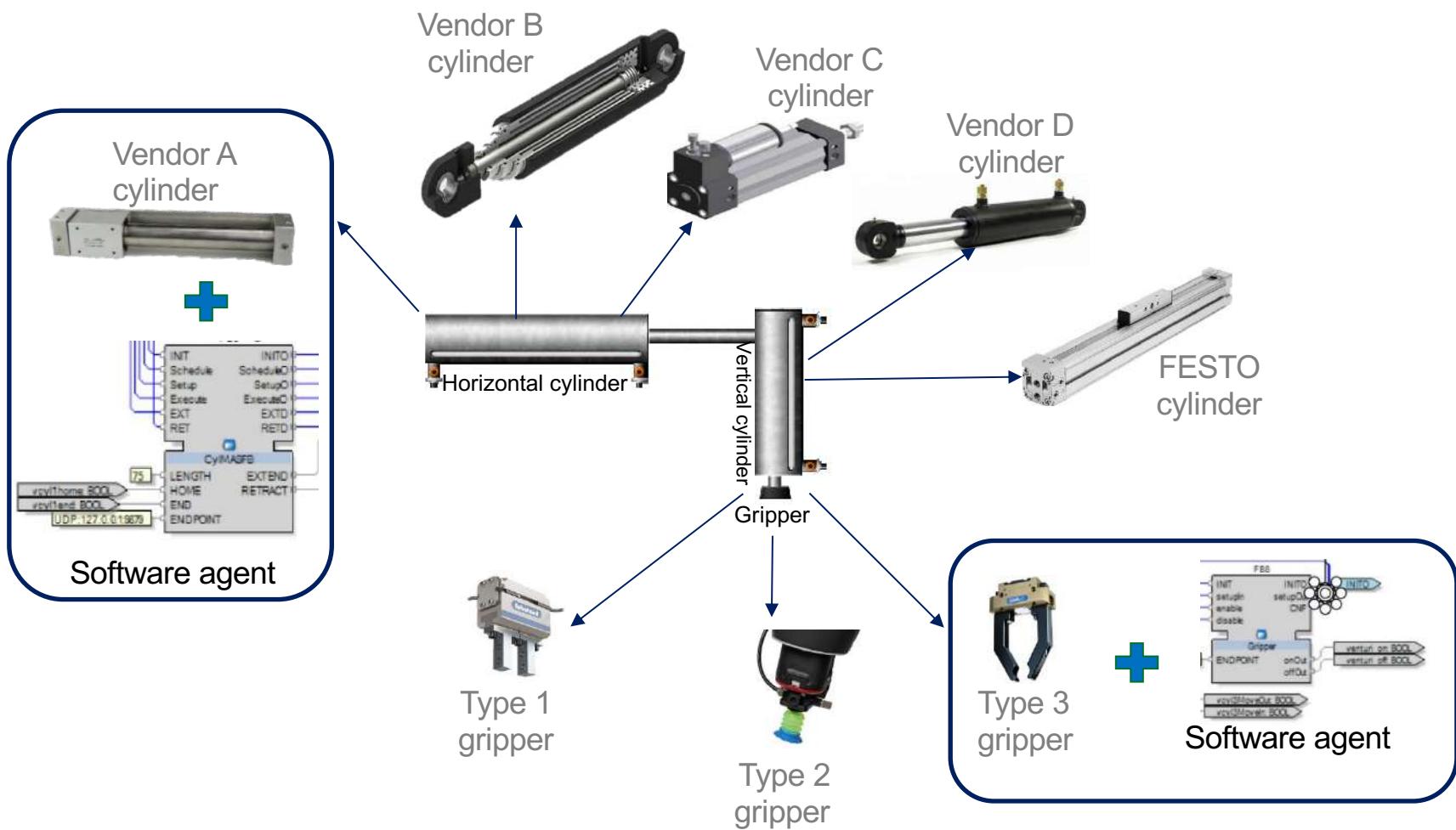


+

Cylinder Software Agent

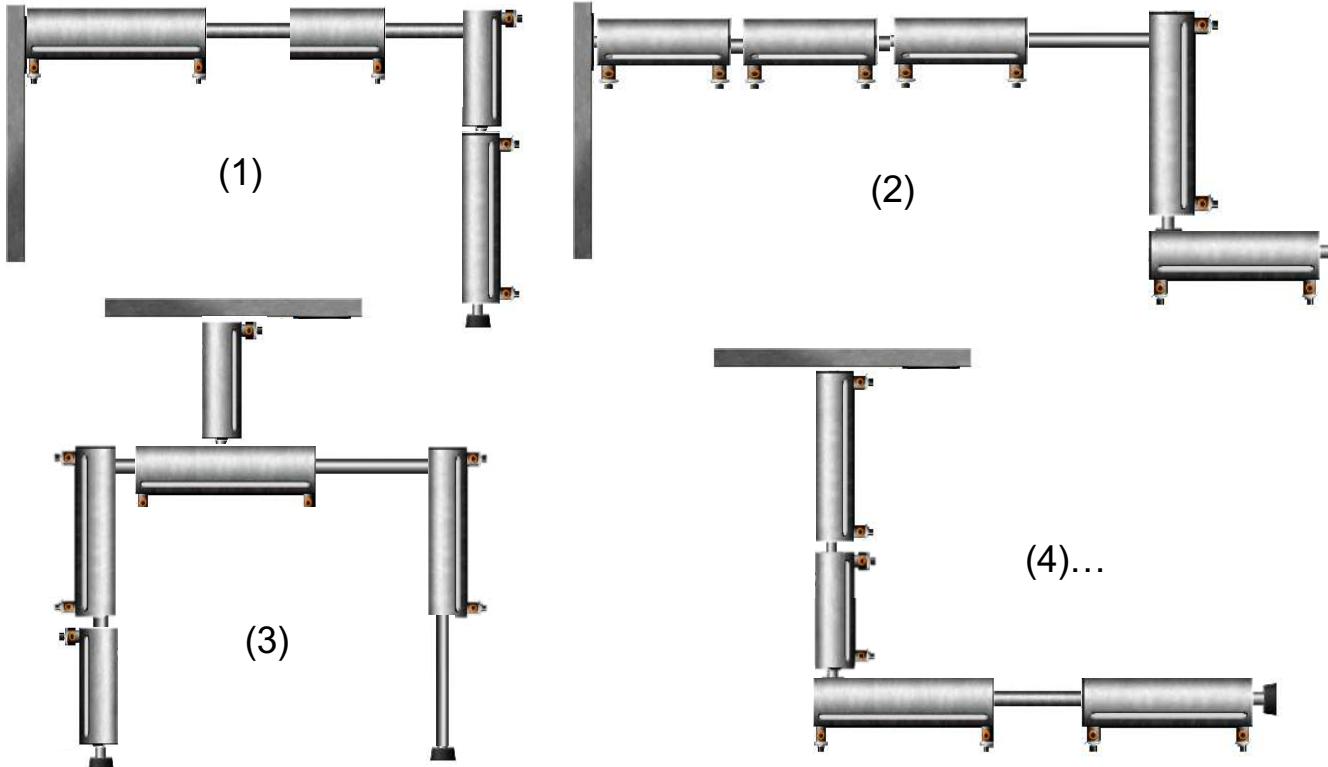


# System integrator perspective: equipment from different vendors

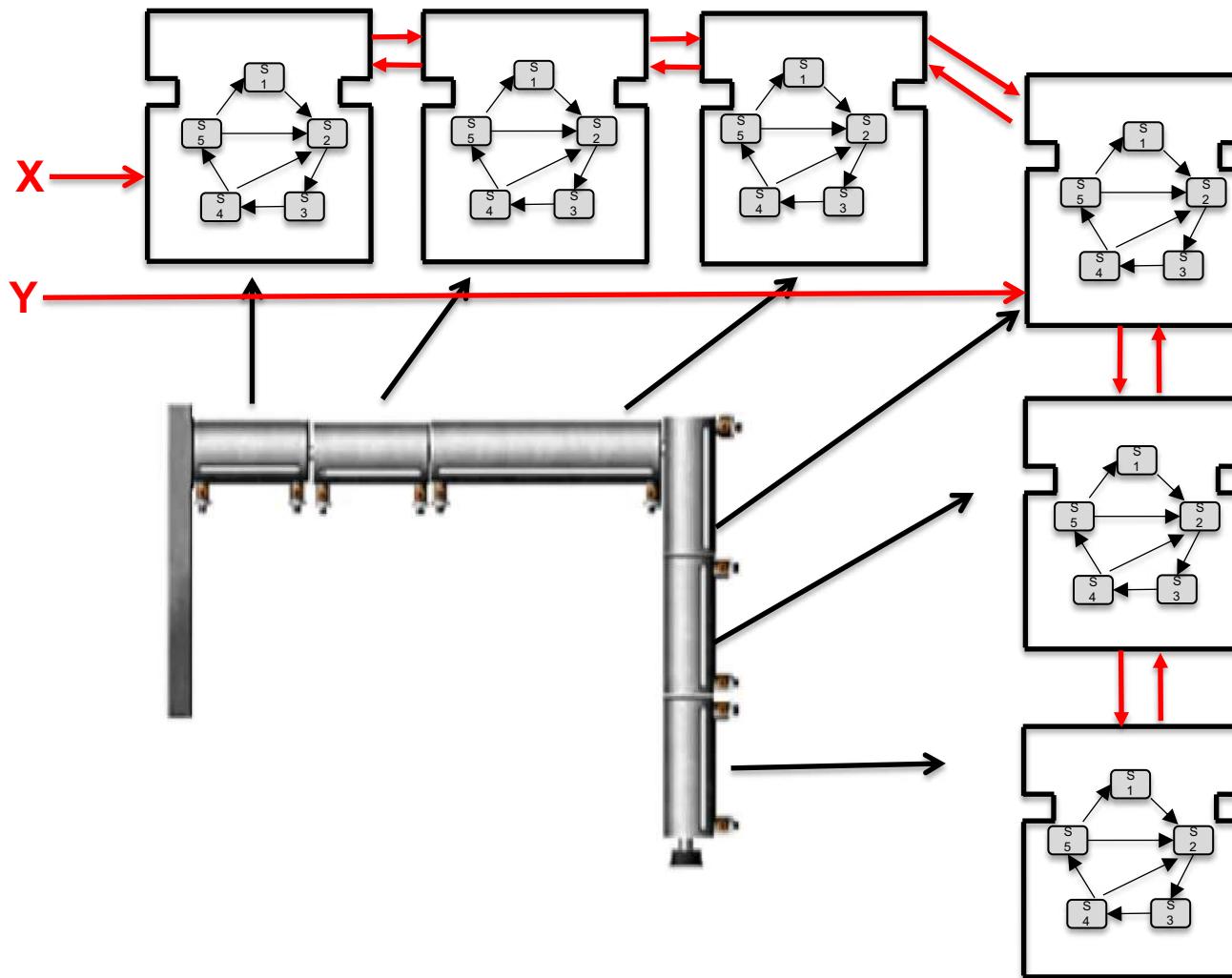


# Different configurations

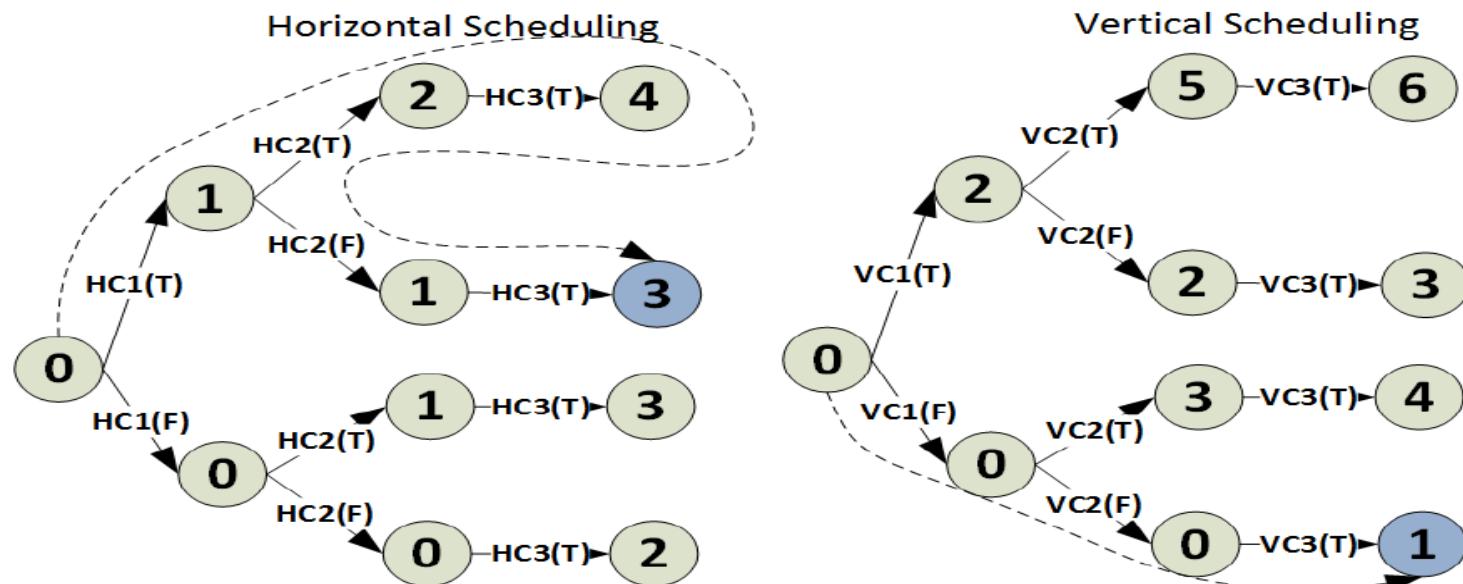
assembled from standard components



# Embedded mechatronic agents do both scheduling and real-time control



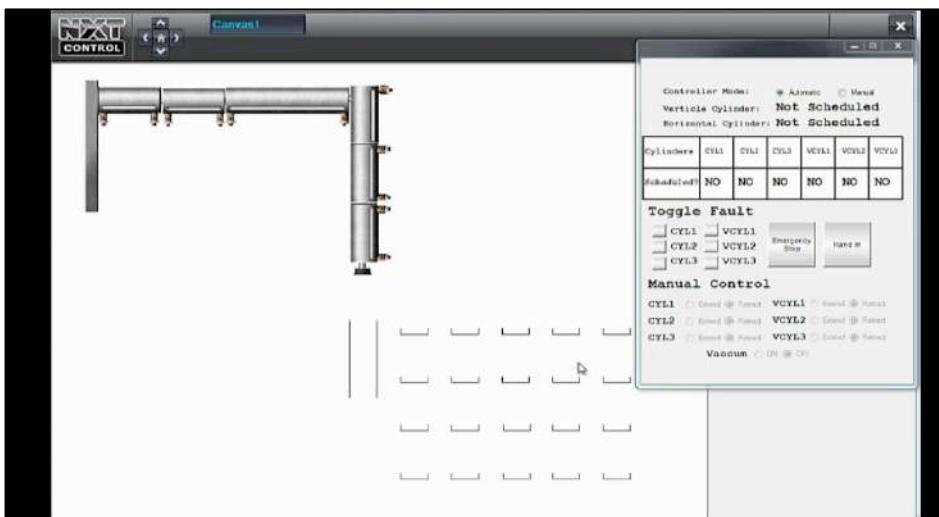
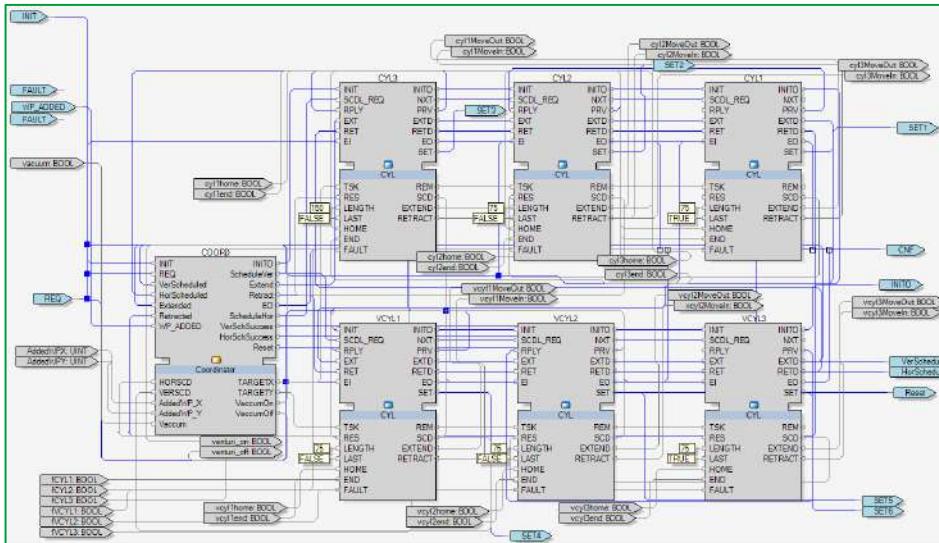
# Search of the Schedule: depth first with backtracking



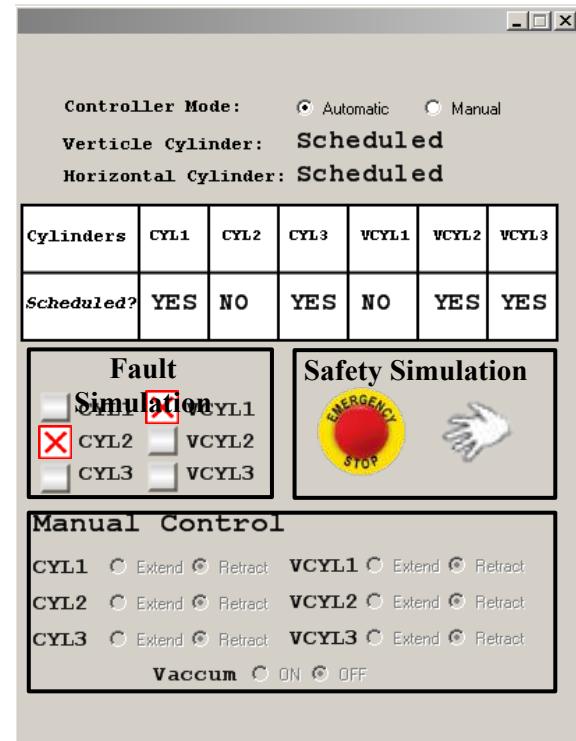
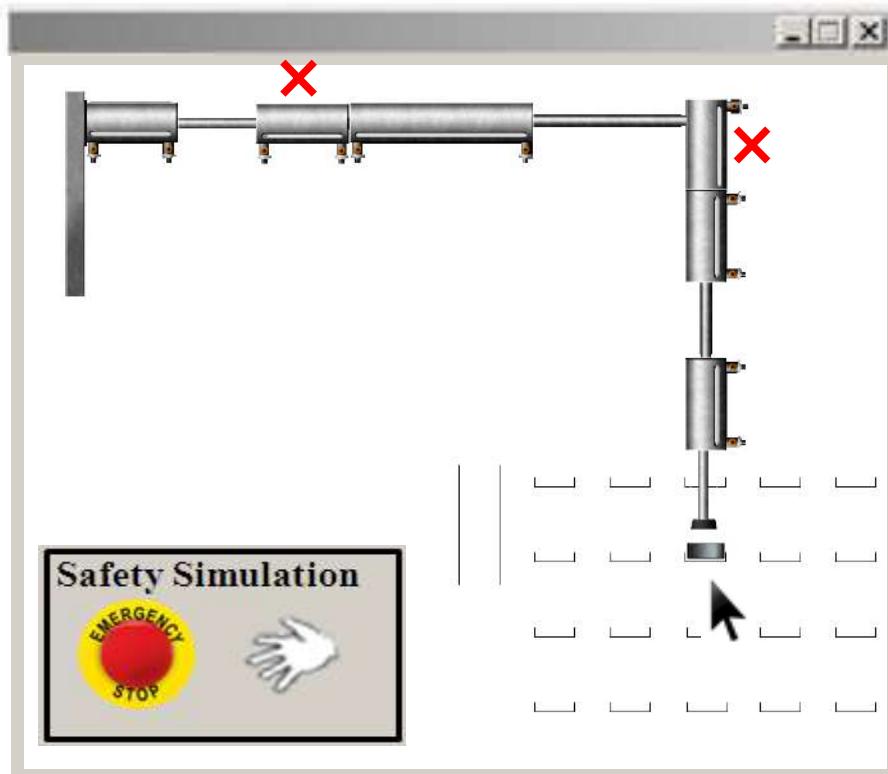
$HC1=1$ ,  $HC2=1$ ,  $HC3=2$   $VC1=2$ ,  $VC2=3$ ,  $VC3=1$ , (T): Operation is True (F): Operation is False, H/VC: Horizontal/Vertical Cylinder

**N** : Units of length reached so far, Example Target Position: ( 3,1)

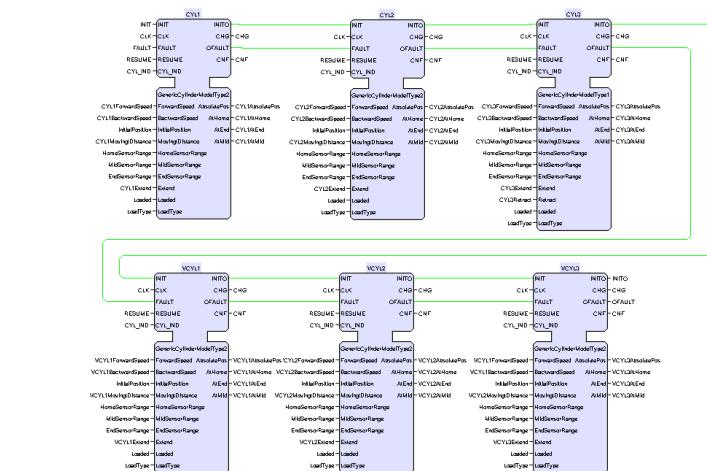
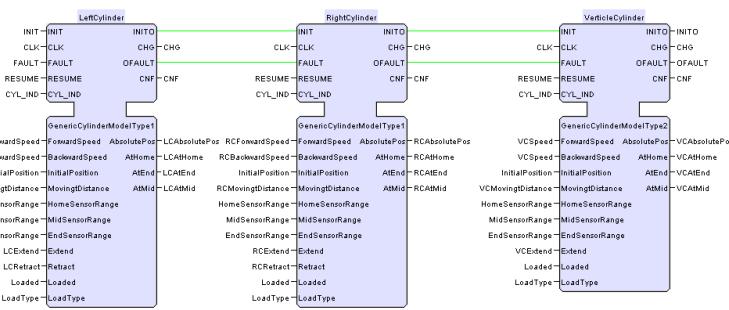
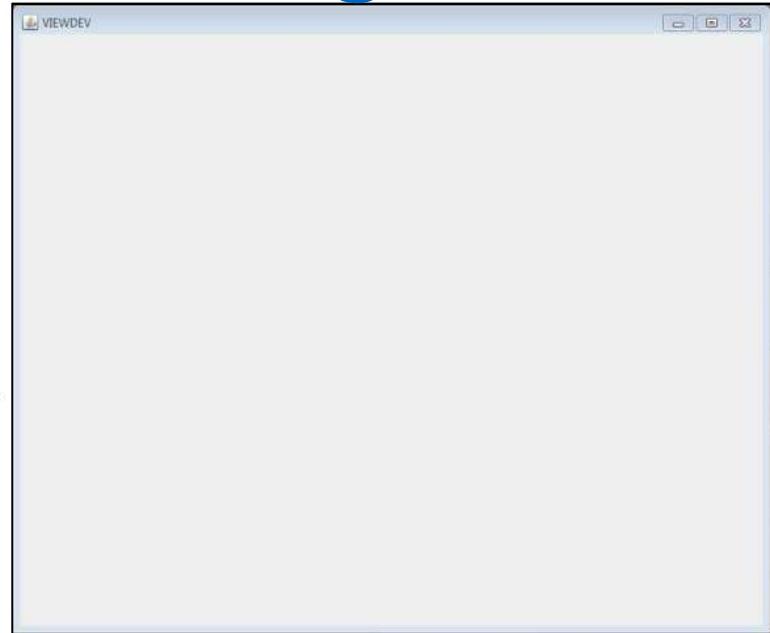
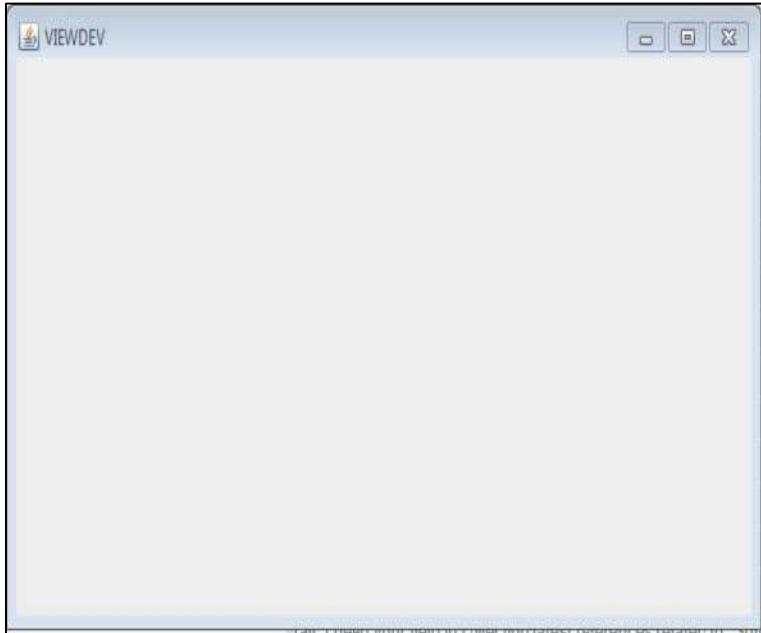
# Embedded mechatronic agents do both scheduling and real-time: in work



# Handling faults on the fly



# Works in different configurations without reprogramming



# Итоги

- Промышленные КФС создают много интересных проблем относящихся к вычислениям, и к моделям
- Для их решения приходится изобретать новые способы программирования и моделирования систем

[Valeriy.Vyatkin@gmail.com](mailto:Valeriy.Vyatkin@gmail.com)