Интернет вещей

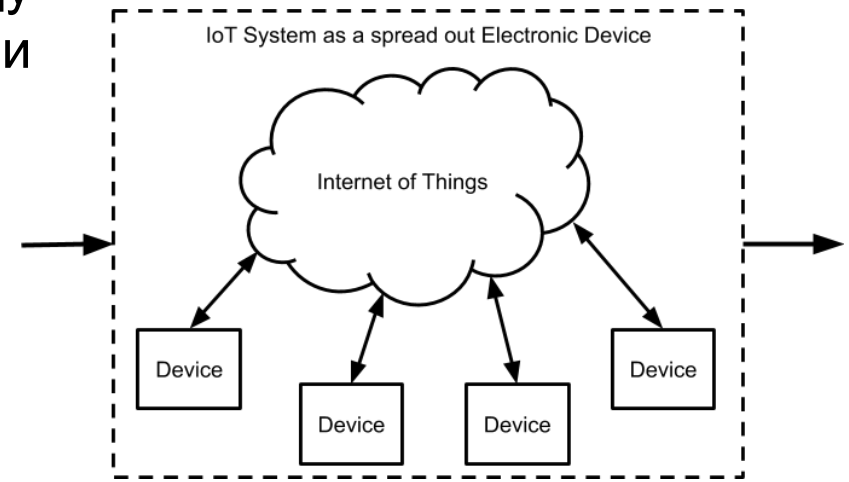
Коммуникация

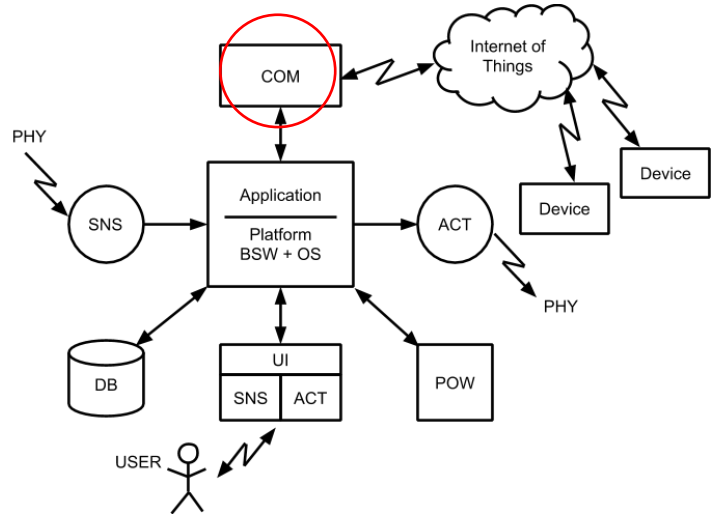
Передача информации

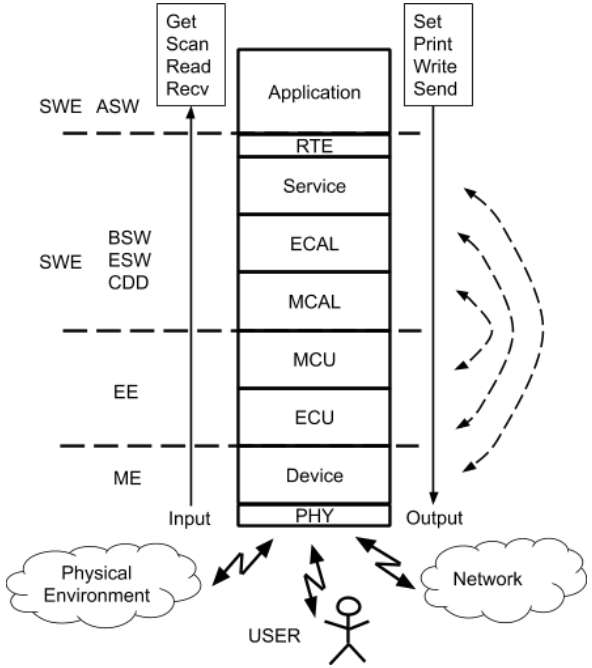
Коммуникация

Обмен информацией между собеседниками

- Понятие коммуникации
- Среда передачи
- Топология сети
- Физический протокол
- Логический протокол
- Интернет/Облако



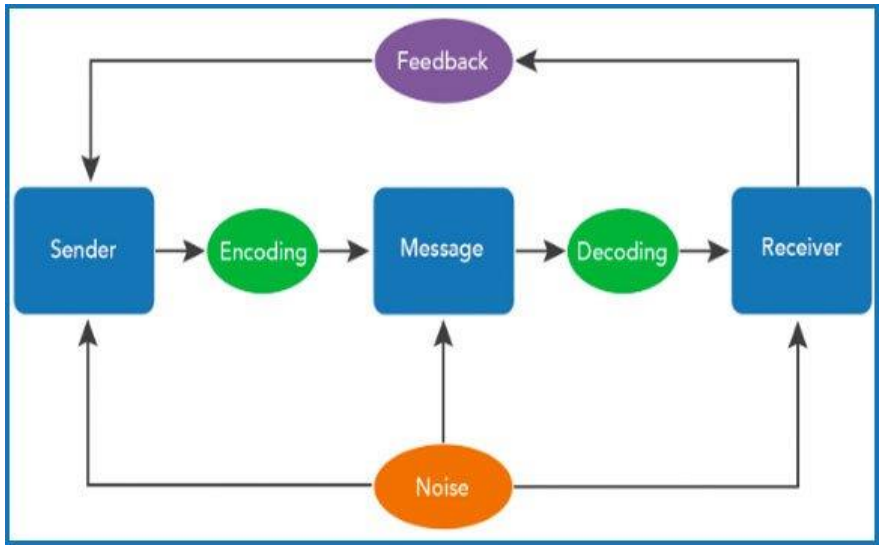




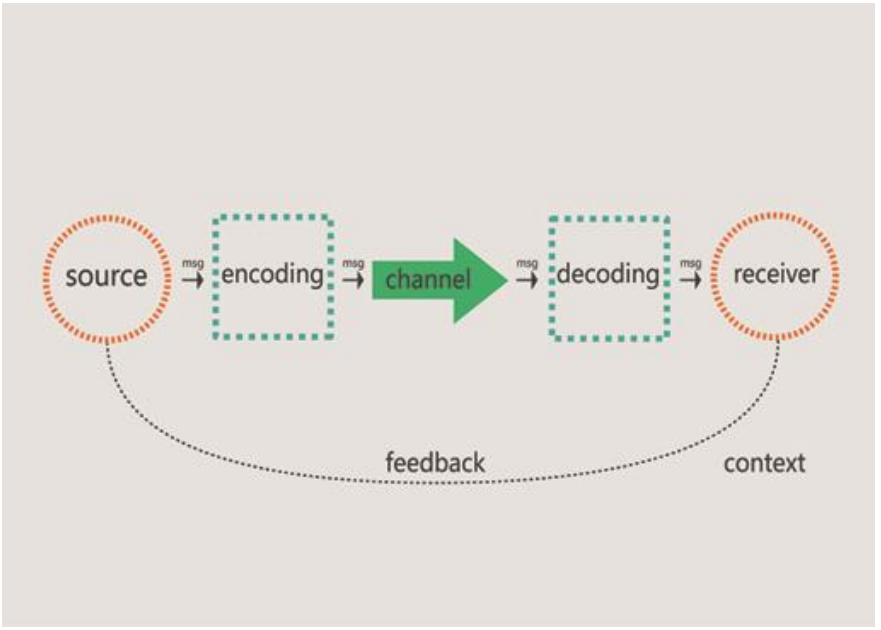
Понятие коммуникации

Обмен информацией между собеседниками

- Сообщение
- Передатчик
- Кодирование
- Канал
- Расшифровка
- Получатель
- Ответ
- Шум



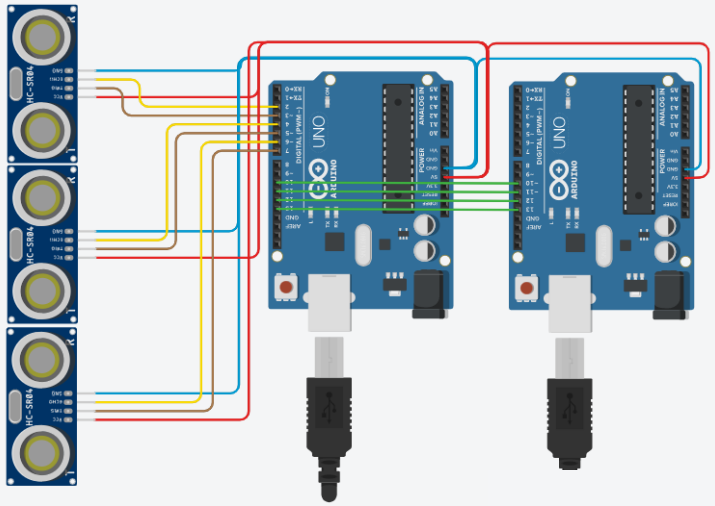
https://learntechit.com/the-process-of-communication/



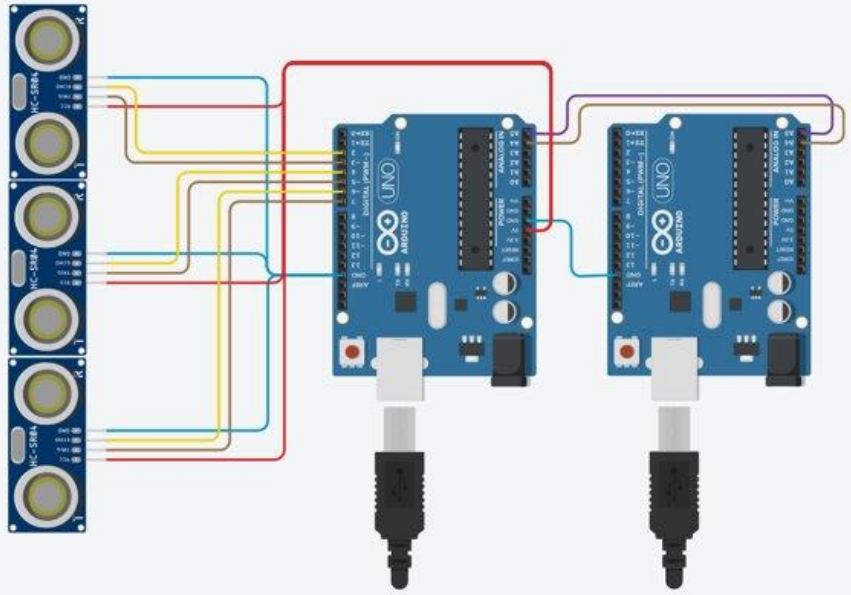
https://www.open.edu/openlearn/ocw/mod/oucontent/view.php?id=87012&section=4

Физический протокол — цифровой ультразвуковой датчик HCS-04

SPI

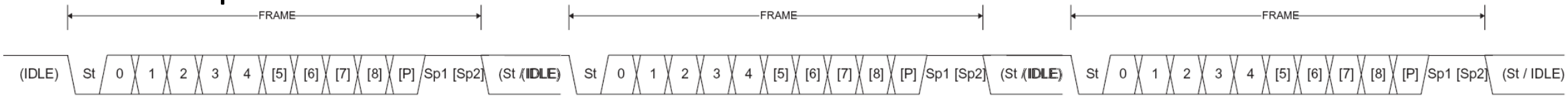


I2C

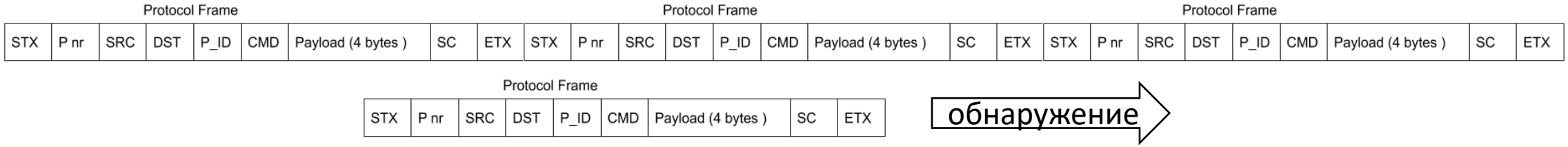


USART - протокол реализации

Физический протокол



Логический протокол

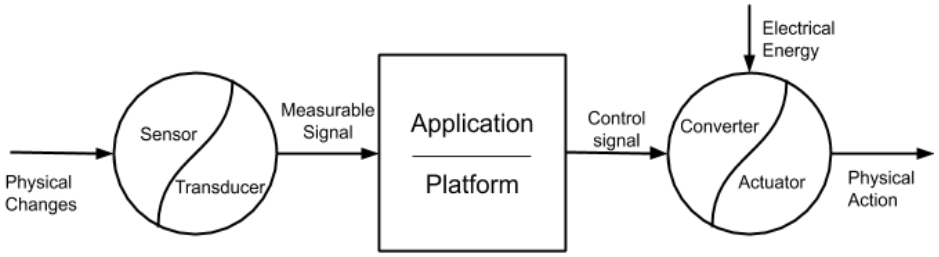


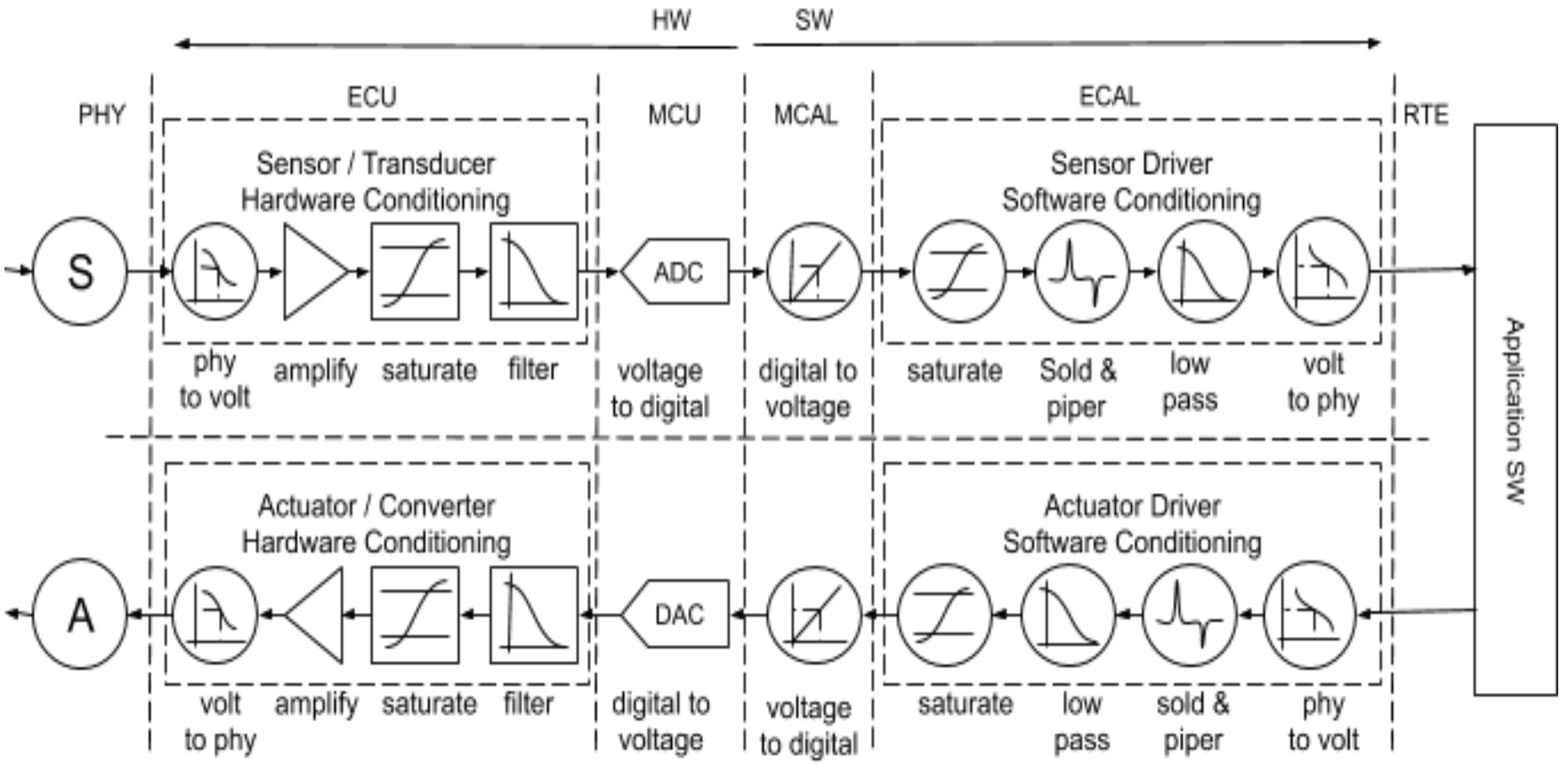
Stx – 0x02
Etx - 0x03
Pnr – счетчик пакетов
SRC – передатчик
DST – получатель
P_id – тип пакета
CMD – команда
Payload – данные пакета
SC – контрольная сумма

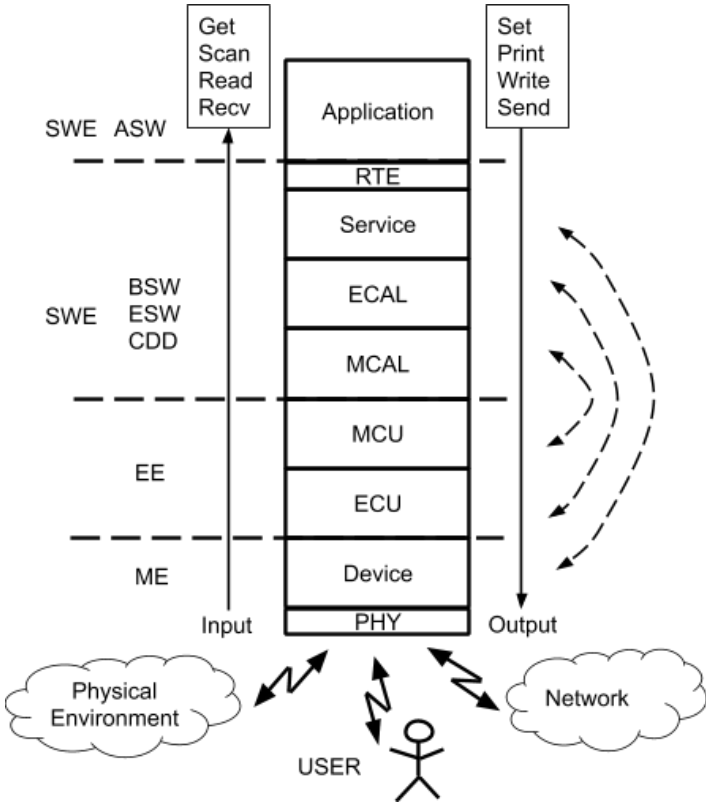
Выпуск
1. Выбор данных
2. Упаковка
3. Создание СЦ
4. Отправка

Прием
1. Сбор байтов
2. Буферизация
3. Проверка
4. Интерпретация данных

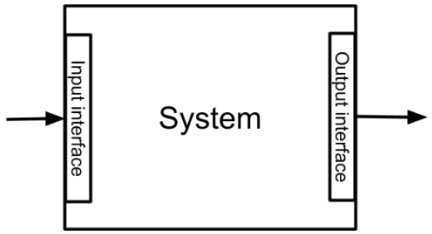
Связь с сигналами

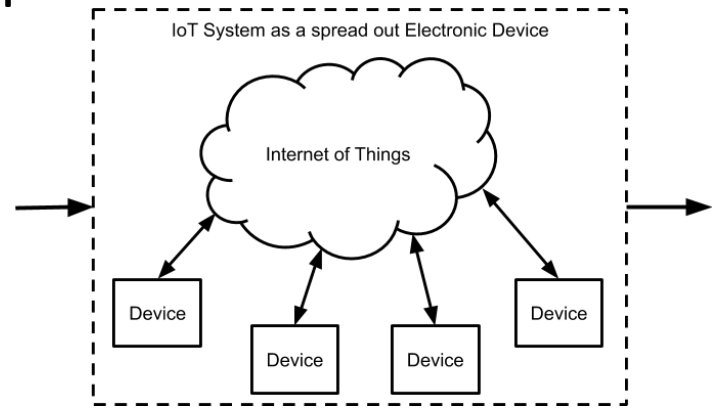


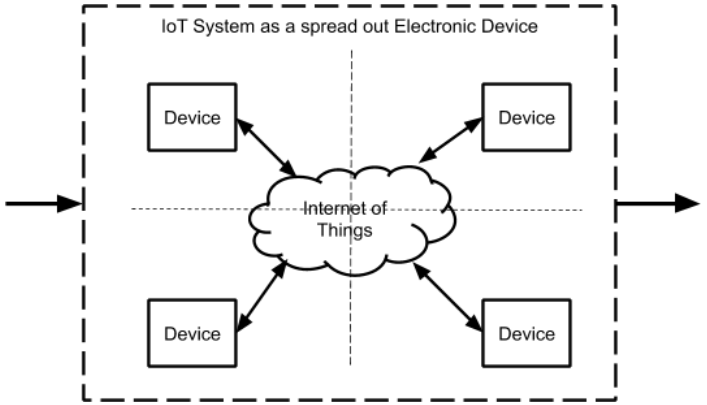




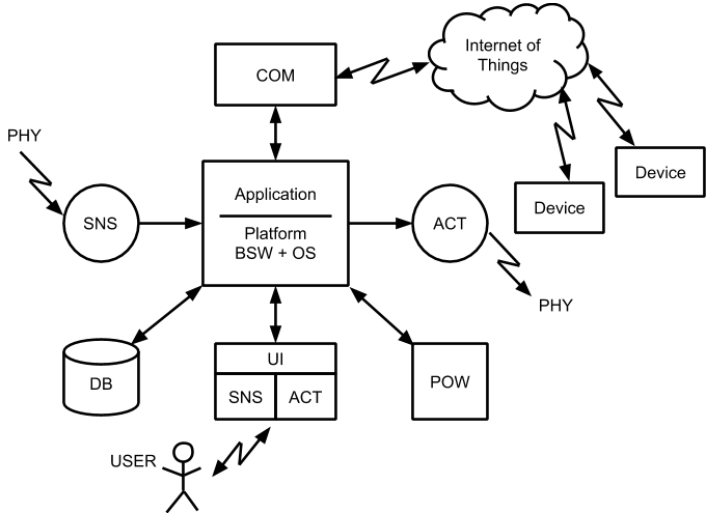
IoT-взаимодействия



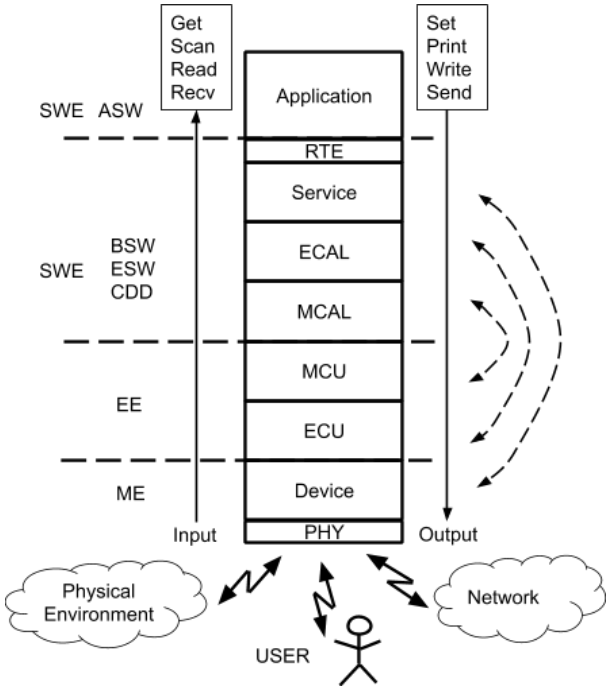




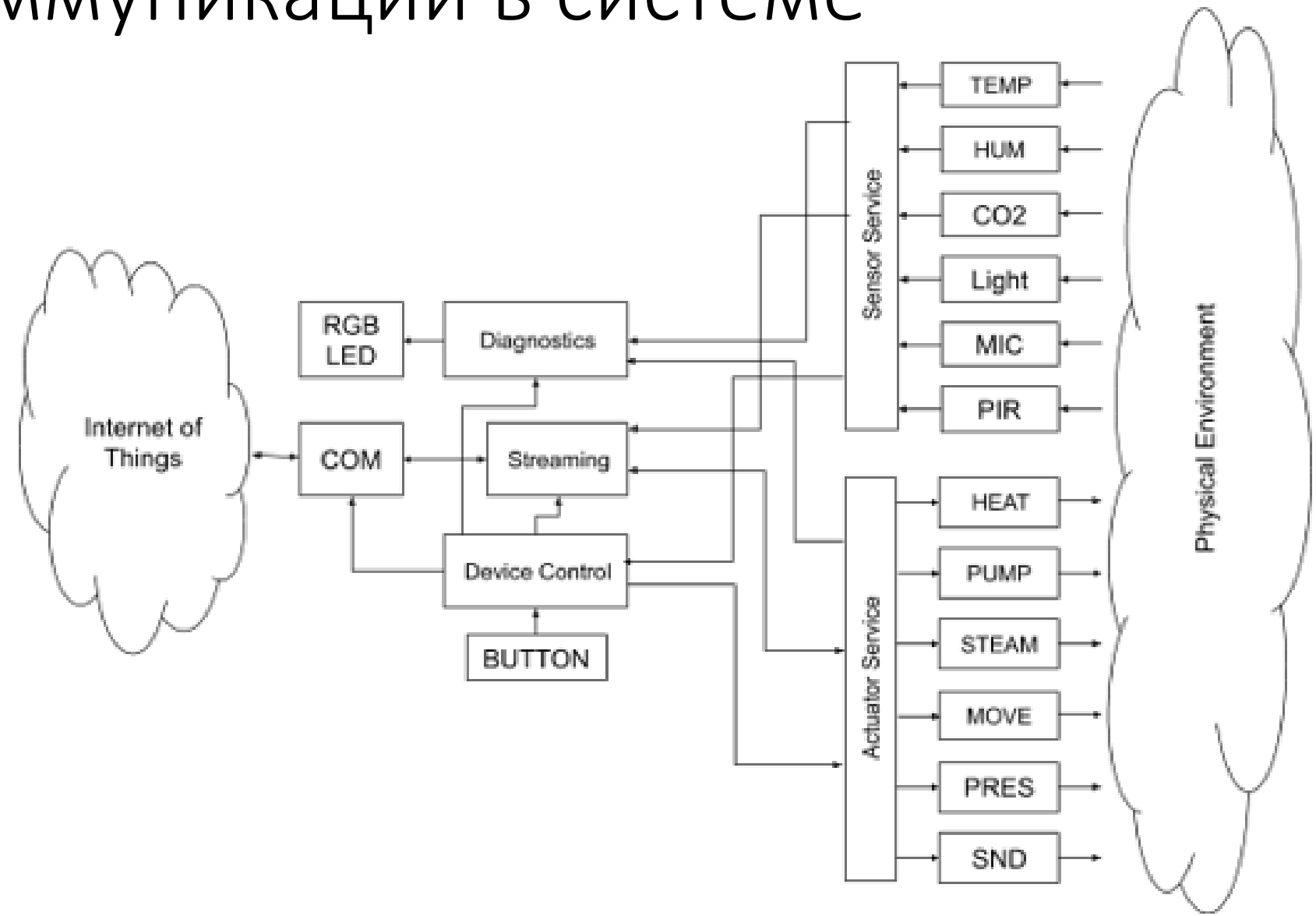
Устройство — Сеть



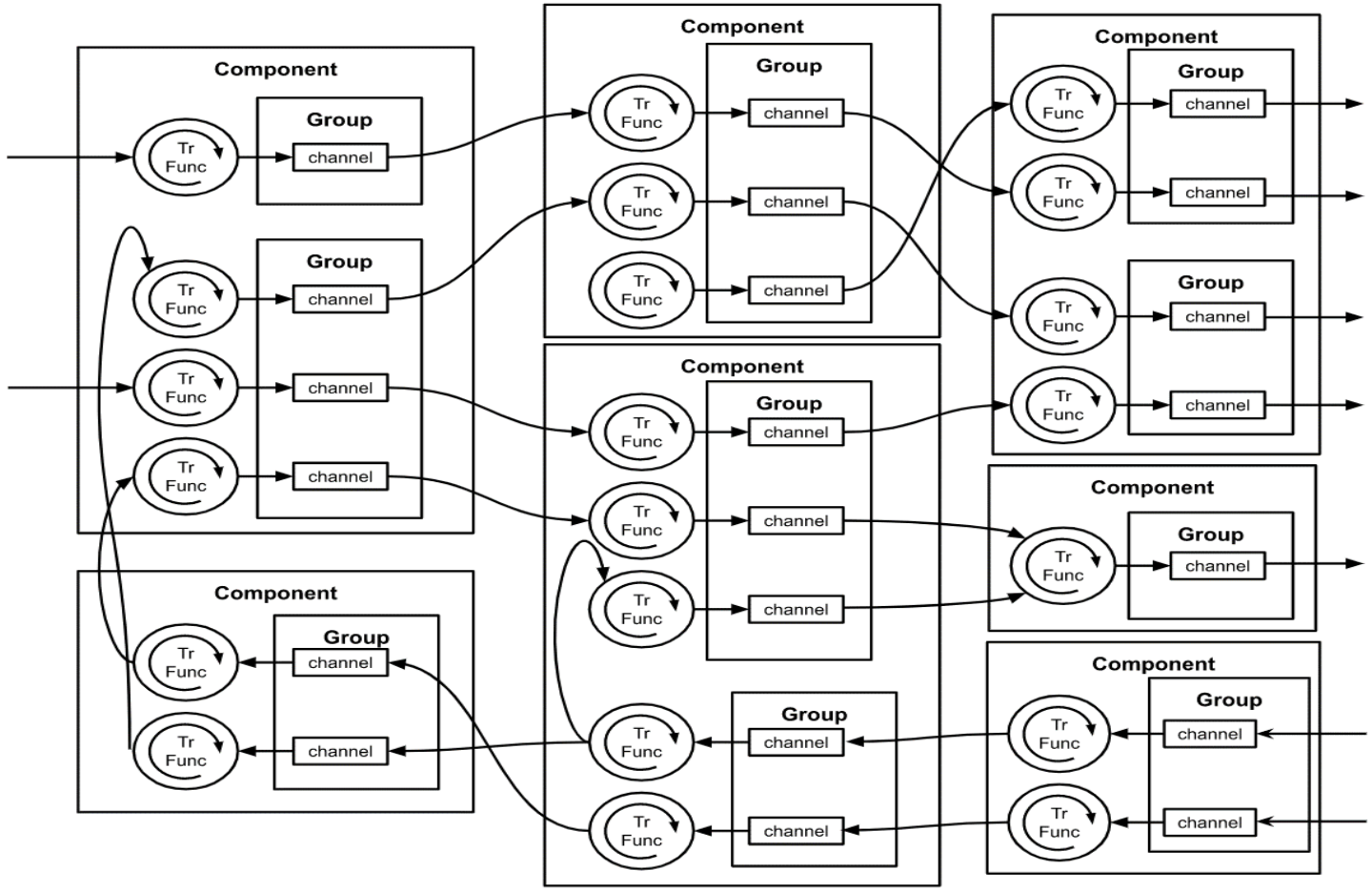
Устройства - Периферийные устройства

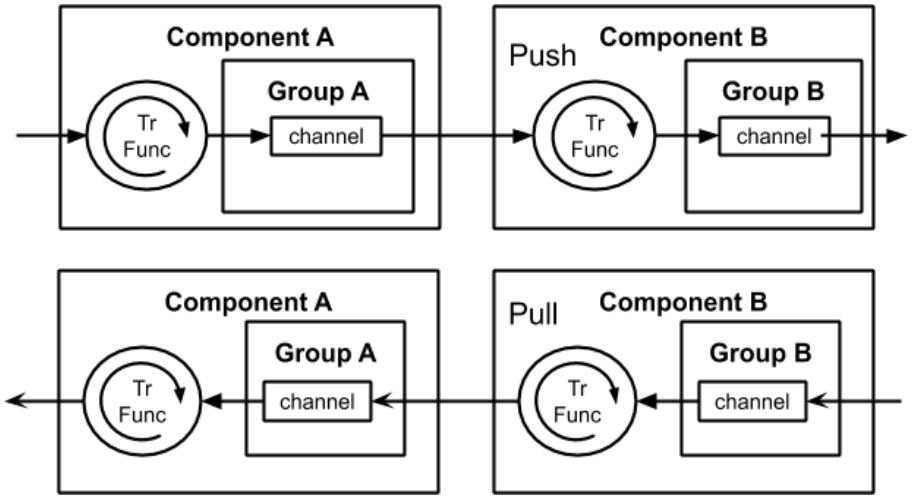


Коммуникации в системе

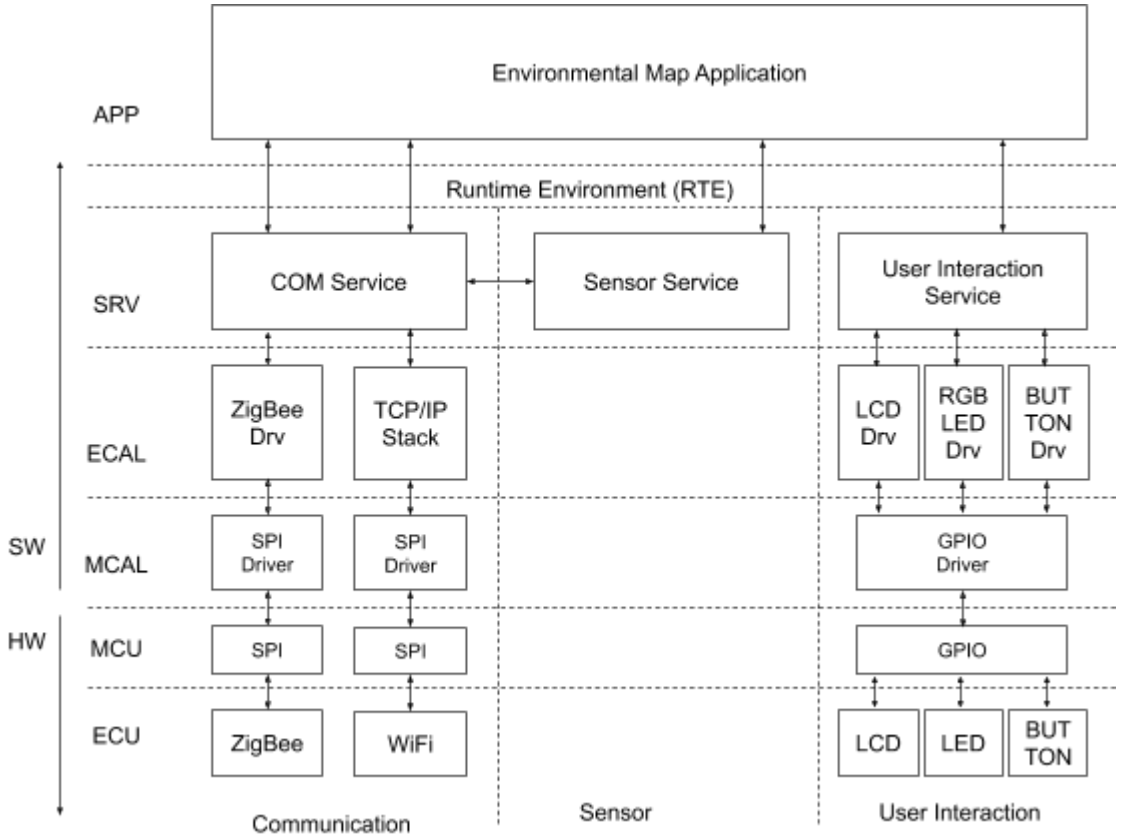


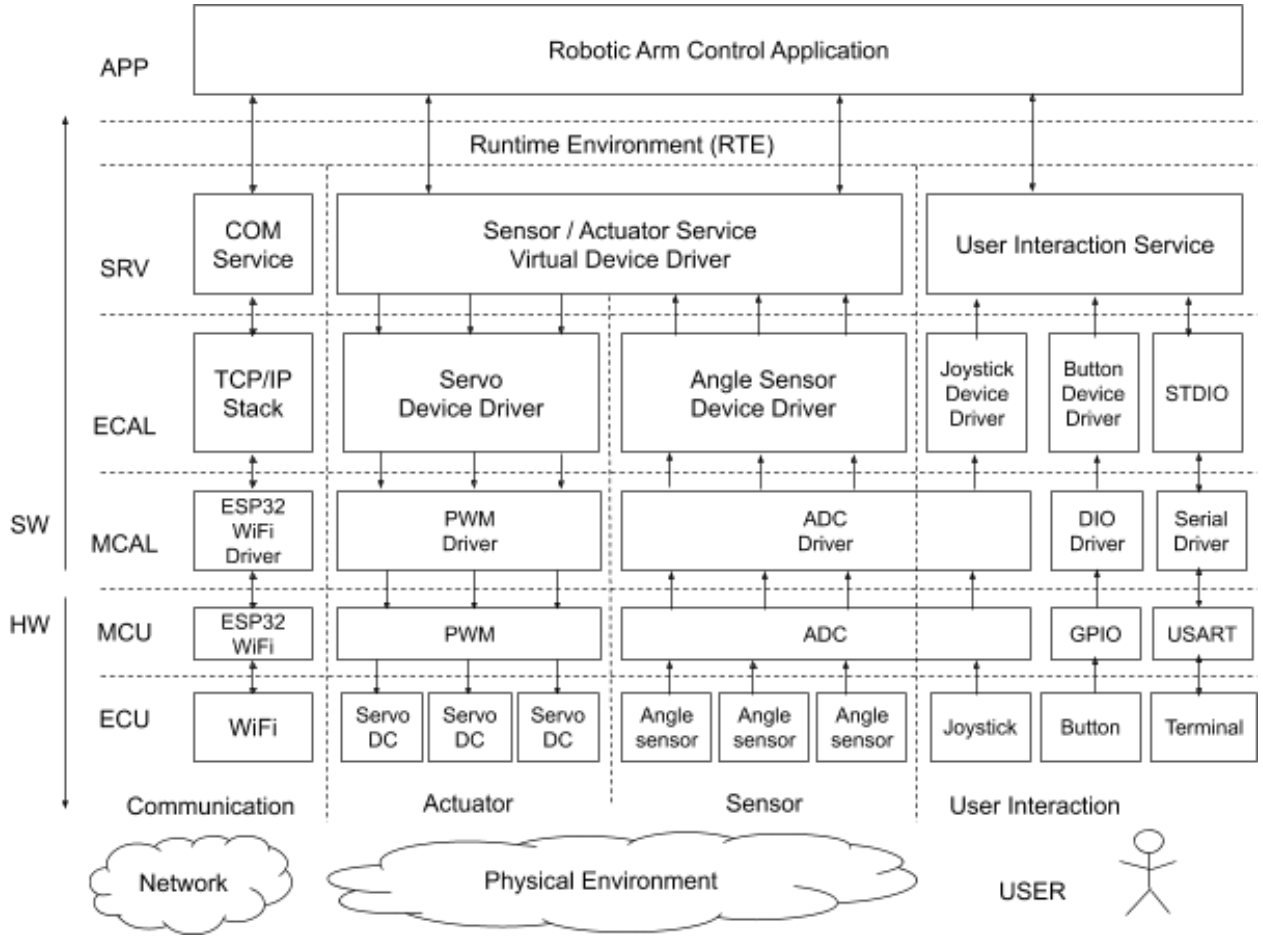
Плоский вид архитектуры системы



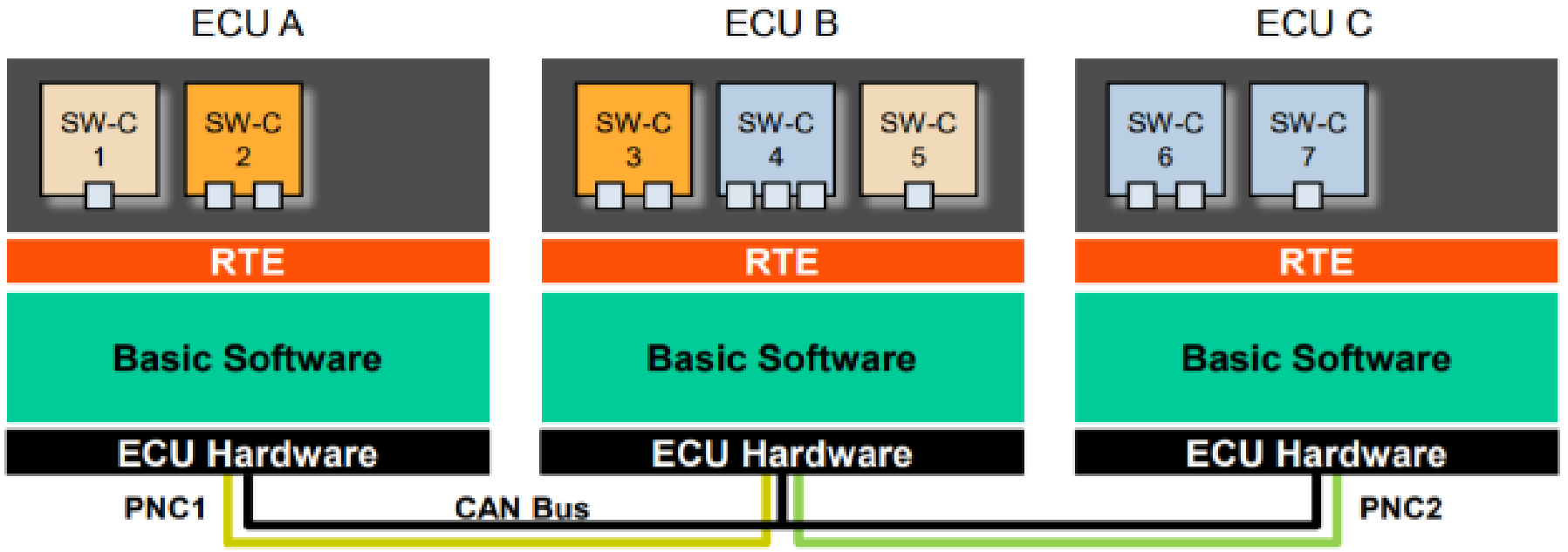


Коммуникации через уровни абстракции



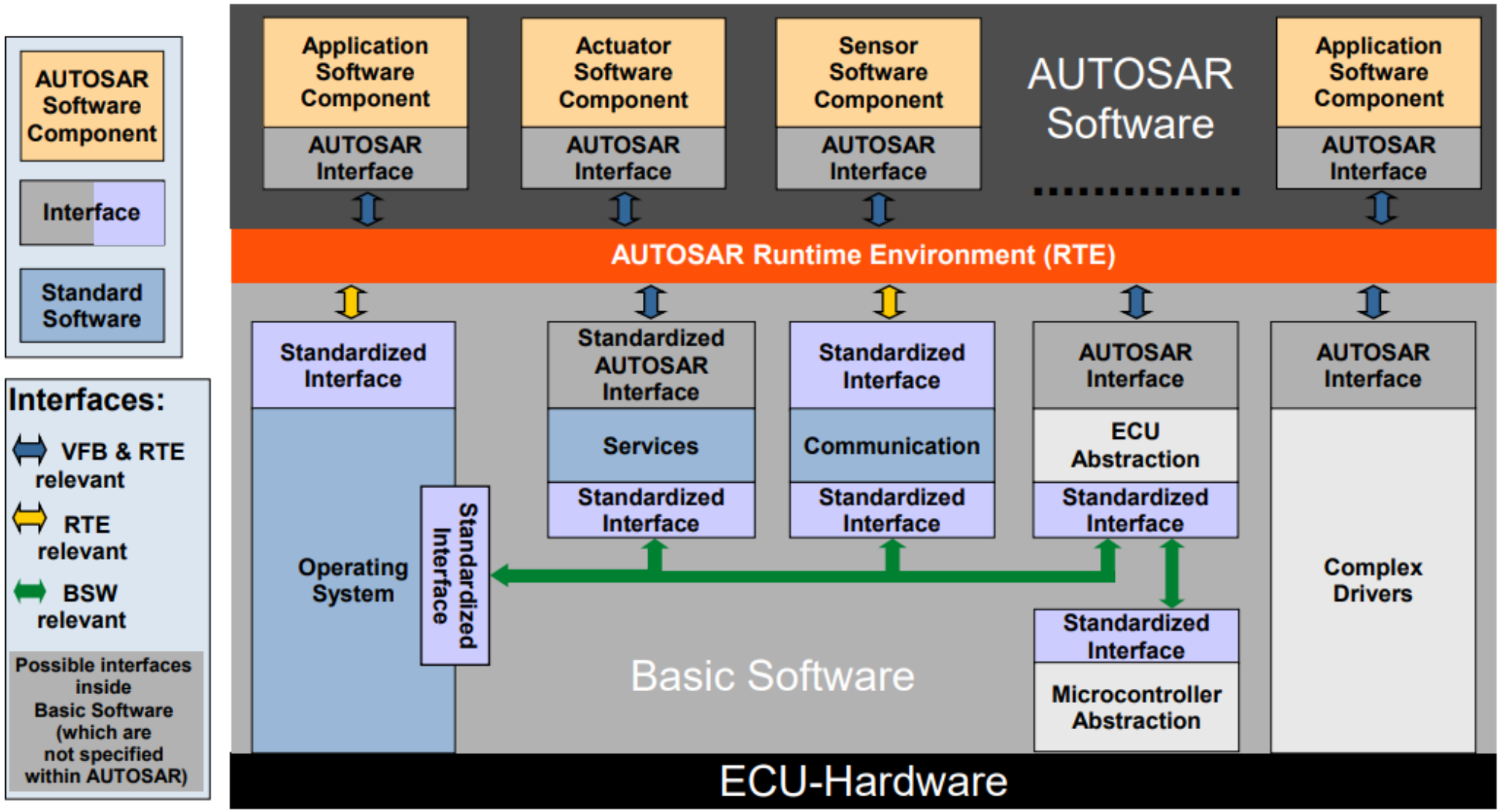


Коммуникации между компонентами



page id: 94ju5

Interfaces
Components and interfaces view (simplified)



Note: This figure is incomplete with respect to the possible interactions between the layers.

https://www.autosar.org/fileadmin/user_upload/standards/classic/4-3/AUTOSAR_EXP_LayeredSoftwareArchitecture.pdf

Interfaces: General Rules
General Interfacing Rules

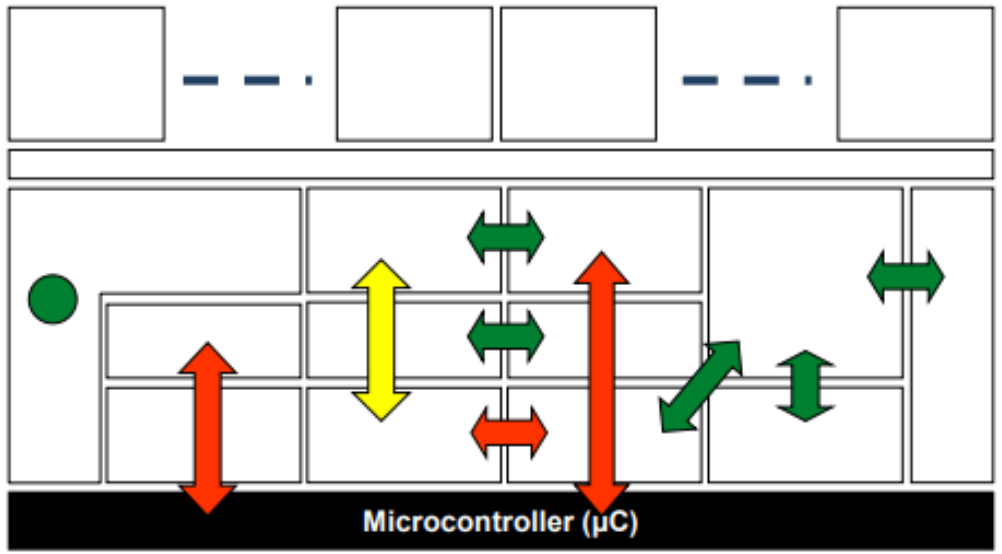
Horizontal Interfaces

Services Layer: horizontal interfaces are allowed
Example: Error Manager saves fault data using the NVRAM manager

ECU Abstraction Layer: horizontal interfaces are allowed

A complex driver may use selected other BSW modules

µC Abstraction Layer: horizontal interfaces are not allowed. Exception: configurable notifications are allowed due to performance reasons.



Vertical Interfaces

One Layer may access all interfaces of the SW layer below

Bypassing of one software layer should be avoided

Bypassing of two or more software layers is not allowed

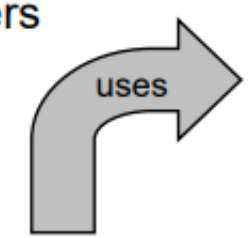
Bypassing of the µC Abstraction Layer is not allowed

A module may access a lower layer module of another layer group (e.g. SPI for external hardware)

All layers may interact with system services.

Interfaces: General Rules
Layer Interaction Matrix

This normative matrix shows the allowed interactions between AUTOSAR Basic Software layers



	System Services / OS	Memory Services	Crypto Services	Communication Services	Off-board Comm. Services	Complex Drivers	I/O Hardware Abstraction	Onboard Device Abstr.	Memory HW Abstraction	Crypto HW Abstraction	Comm. HW Abstraction*	Microcontroller Drivers	Memory Drivers	Crypto Drivers	Communication Drivers*	I/O Drivers	Microcontroller Hardware
SW Components / RTE	✓	✓	✓	✓	✓	✓	✓	×	×	×	×	×	×	×	×	×	×
System Services / OS	✓	✓	✓	✓	✓	Δ	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Memory Services	✓	✓	✓	×	×	Δ	×	×	✓	×	×	×	×	×	×	×	×
Crypto Services	✓	✓	✓	×	×	Δ	×	×	×	✓	×	×	×	×	×	×	×
Communication Services	✓	✓	✓	✓	✓	Δ	×	×	×	×	✓	×	×	×	×	×	×
Off-board Comm. Services	✓	✓	✓	✓	✓	Δ	×	×	×	×	✓	×	×	×	×	×	×
Complex Drivers	restricted access -> see the following two slides																
I/O Hardware Abstraction	✓	×	✓	×	×	×	✓	✓	×	×	✓	✓	×	×	✓	✓	×
Onboard Device Abstr.	✓	×	×	×	×	×	×	✓	×	×	✓	✓	×	×	✓	✓	×
Memory HW Abstraction	✓	✓	×	×	×	×	×	✓	✓	×	✓	×	✓	×	✓	×	×
Crypto HW Abstraction	✓	✓	✓	×	×	×	×	✓	×	✓	×	×	×	×	×	✓	×
Comm. HW Abstraction*	✓	×	×	✓	✓	×	×	✓	×	×	✓	×	×	×	✓	✓	×
Microcontroller Drivers	✓	×	×	×	×	×	Δ	Δ	×	Δ	×	Δ	×	×	×	Δ	✓
Memory Drivers	✓	×	×	×	×	×	×	×	Δ	×	×	×	×	×	×	×	✓
Crypto Drivers	✓	×	×	×	×	×	×	×	×	Δ	×	×	×	×	×	×	✓
Communication Drivers*	✓	×	×	×	×	×	×	Δ	×	×	Δ	×	×	×	×	✓	✓
I/O Drivers	✓	×	×	×	×	×	Δ	Δ	×	Δ	×	Δ	×	×	×	Δ	✓

*: includes wired and wireless communication

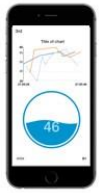
✓ allowed to use
× not allowed to use
Δ restricted use (callback only)

The matrix is read **row-wise**:
Example: "I/O Drivers are allowed to use System Services and Hardware, but no other layers".

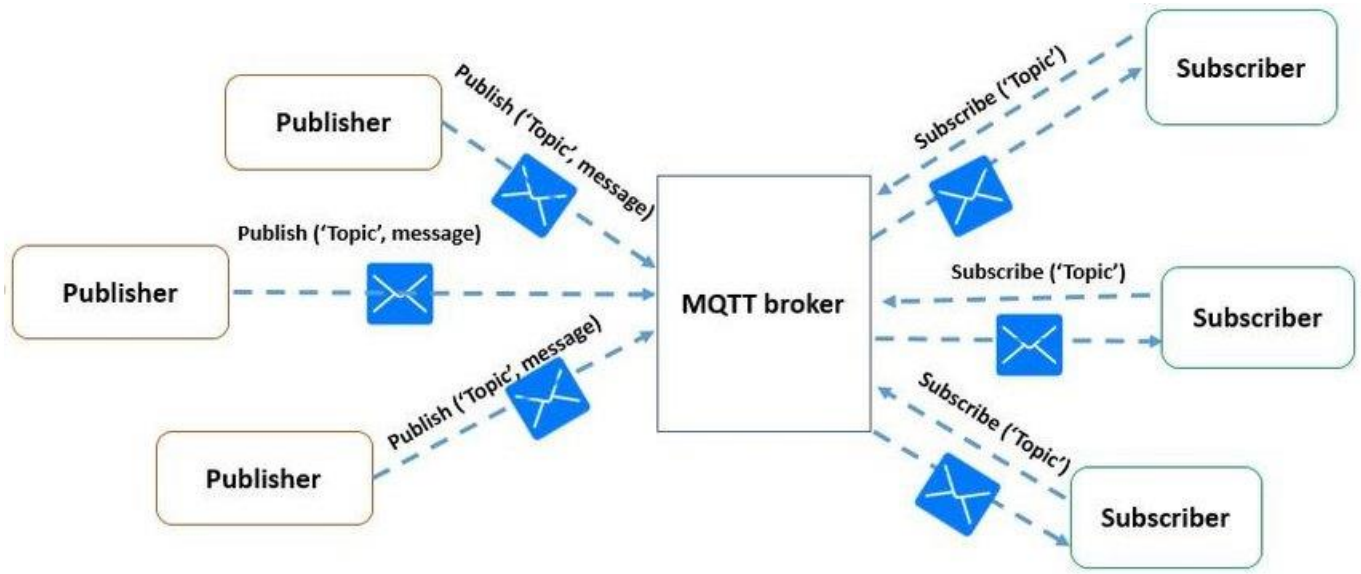
(gray background indicates "non-Basic Software" layers)

IoT с помощью MQTT

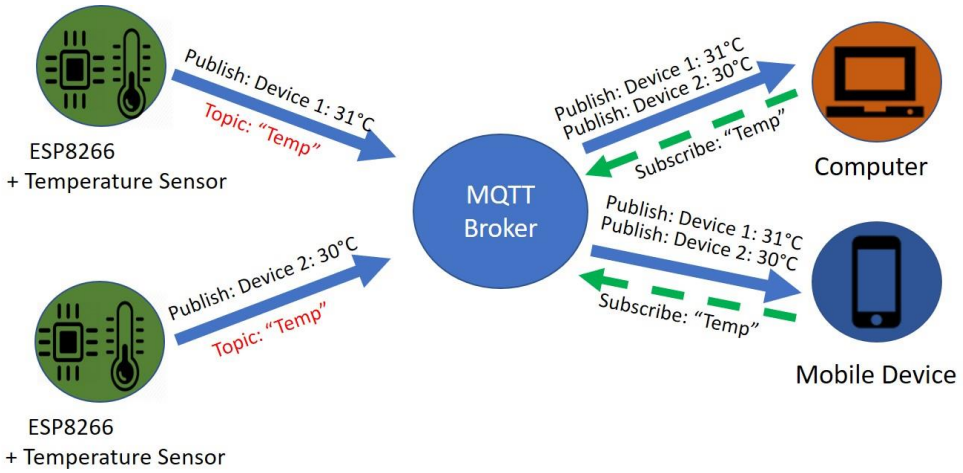
Timeline full of savage comeback tweets

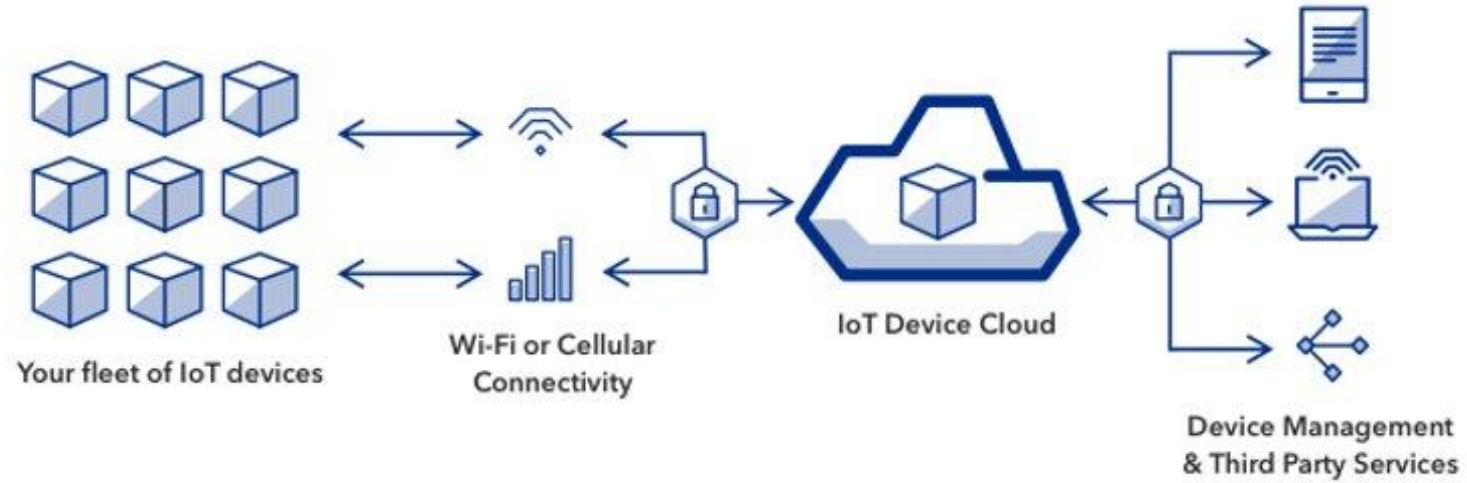


Dashboard full of information

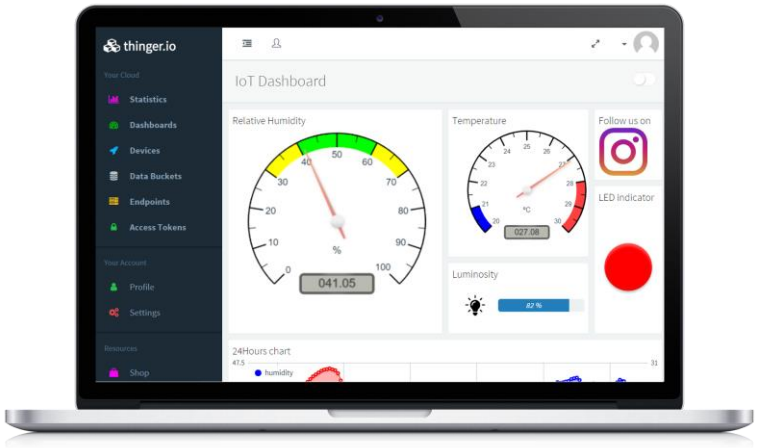


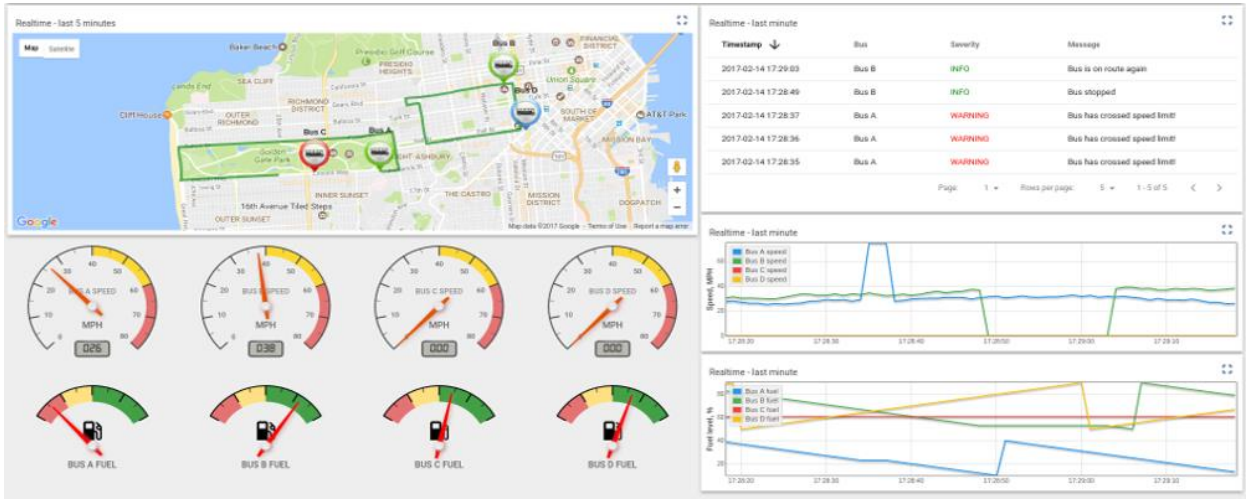
IoT с помощью MQTT





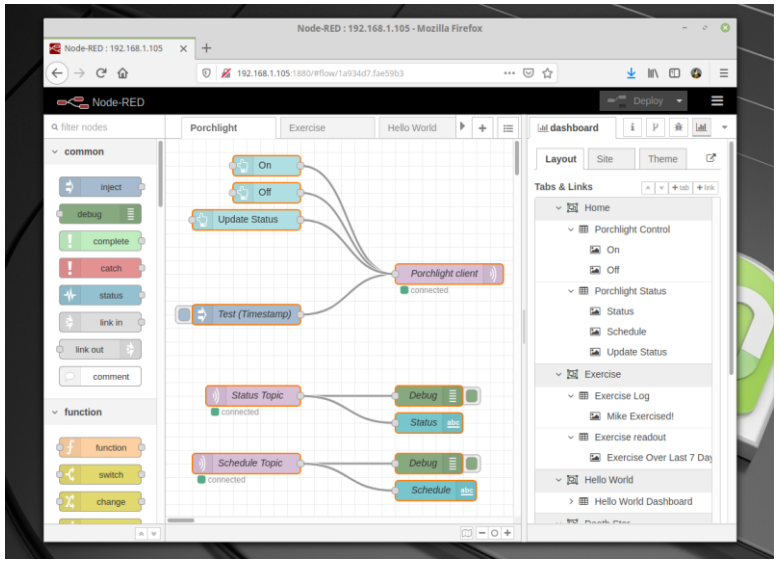
MQTT – Мониторинг (приборная панель)

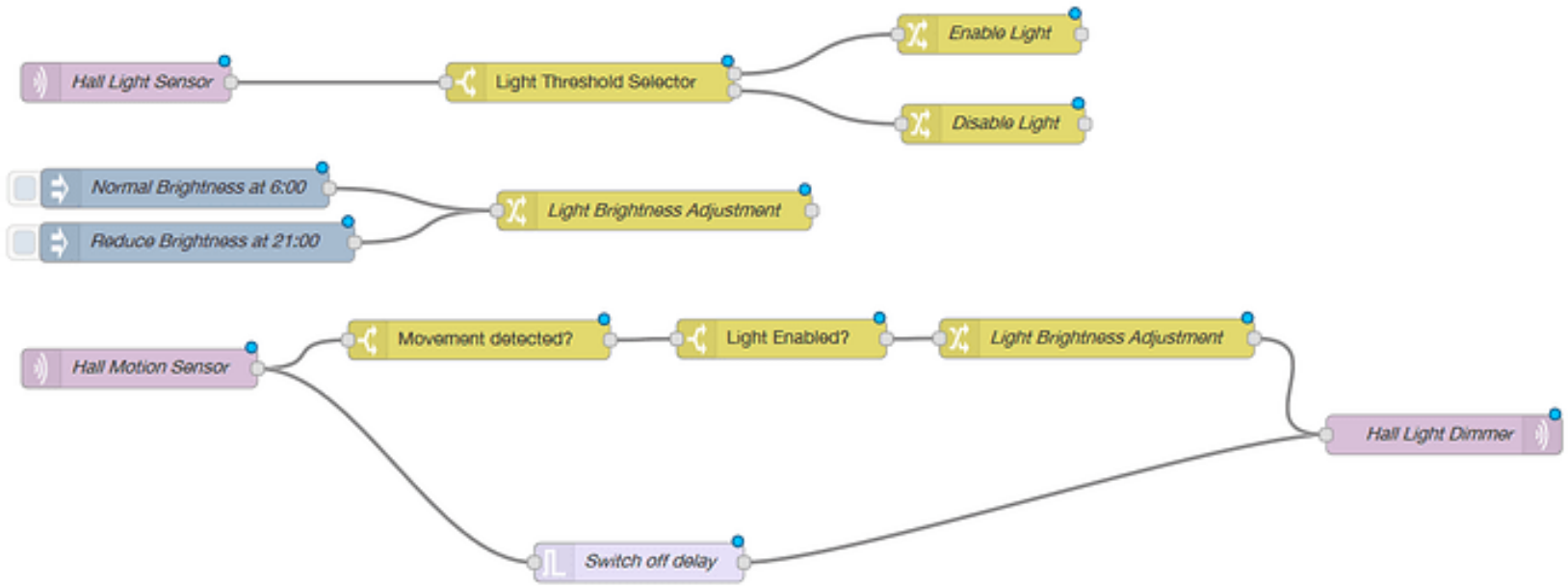




https://thingsboard.io/

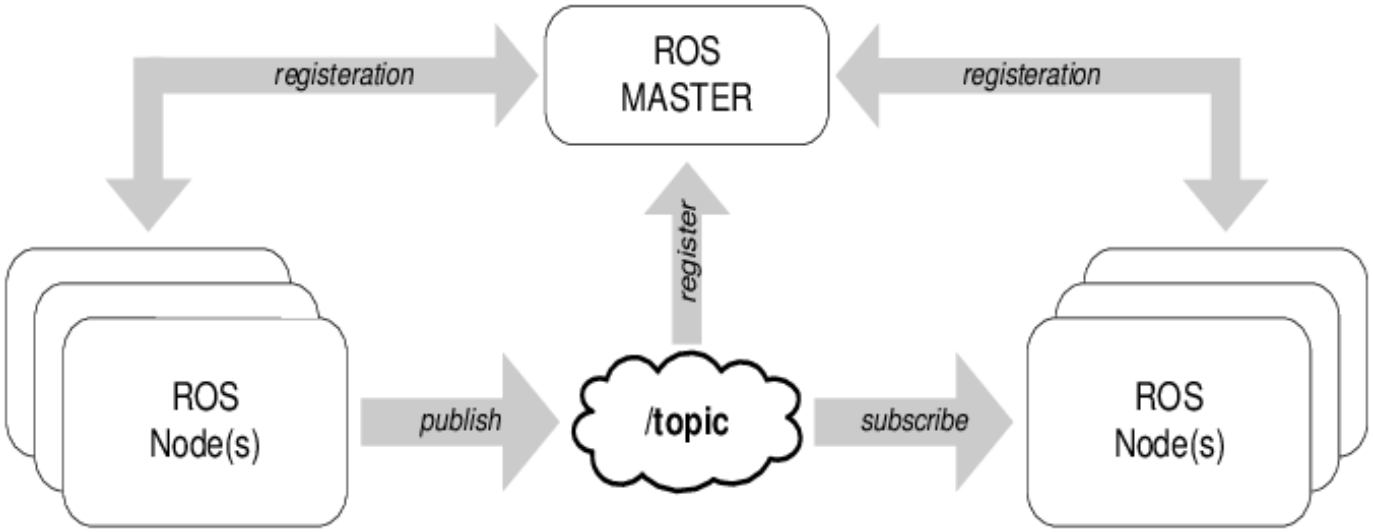
MQTT — Управление (движок правил)





ROS - связь между узлами

https://en.wikipedia.org/wiki/Robot_Operating_System



https://www.google.com/search?q=ros+communication+between+nodes&sxsrf=ALeKk00mbEhAN3jDQ7M6LnF7WAaA9z2DtA:1605018111013&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjeoPWylvjsAhUliIsKHazdDNgQ_AUoAXoECBAQAw&biw=1920&bih=880#imgrc=PapuNi1aYmMdpM

ROS Industrial - IIoT

ROS-Industrial — это проект с открытым исходным кодом, который расширяет расширенные возможности ROS для автоматизации производства и робототехники. Репозиторий ROS-Industrial включает интерфейсы для обычных промышленных манипуляторов, захватов, датчиков и сетей устройств. Он также предоставляет программные библиотеки для автоматической 2D/3D-калибровки датчиков, планирования траектории/движения процесса, такие приложения, как Scan-N-Plan, инструменты разработчика, такие как плагин Qt Creator ROS, и учебную программу, специально предназначенную для нужд производителей. ROS-I поддерживается международным консорциумом представителей промышленности и исследователей.

https://rosindustrial.org/about/description/