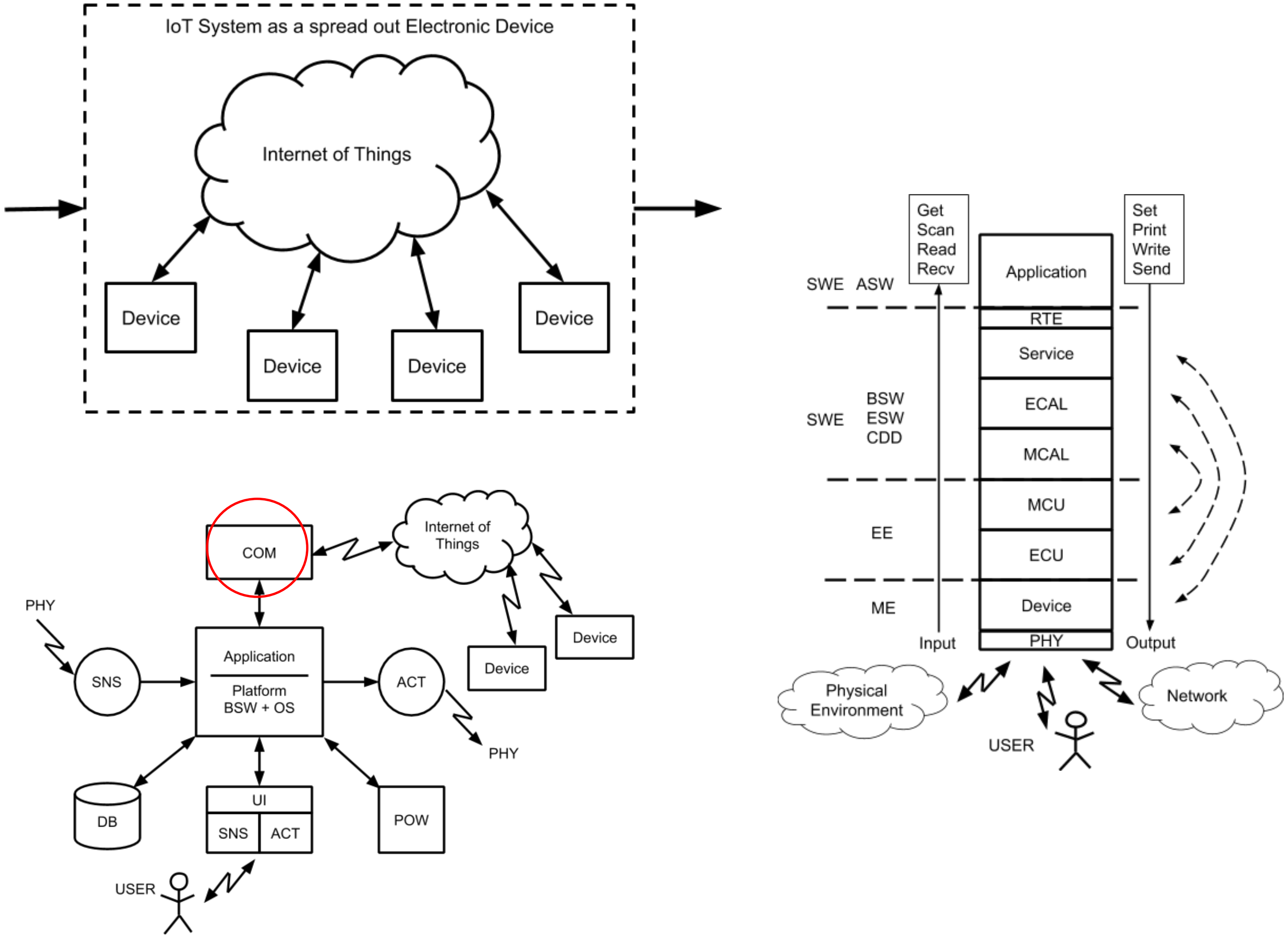
Internetul Lucrurilor

Transfer Informație

Comunicare

Schimb de informație între interlocutori

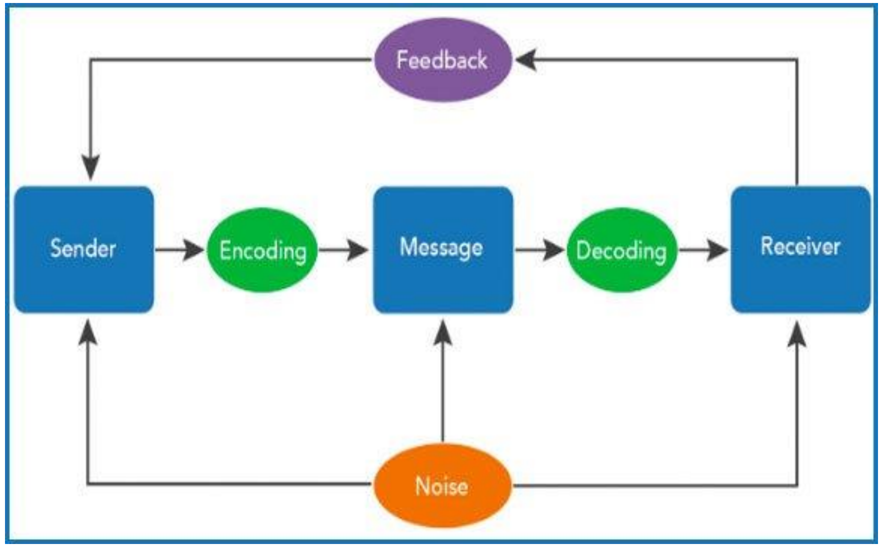
- Notiune de comunicare
- Mediu de transmise
- Topologie retea
- Protocol fizic
- Protocol logic
- Internet/Clouding



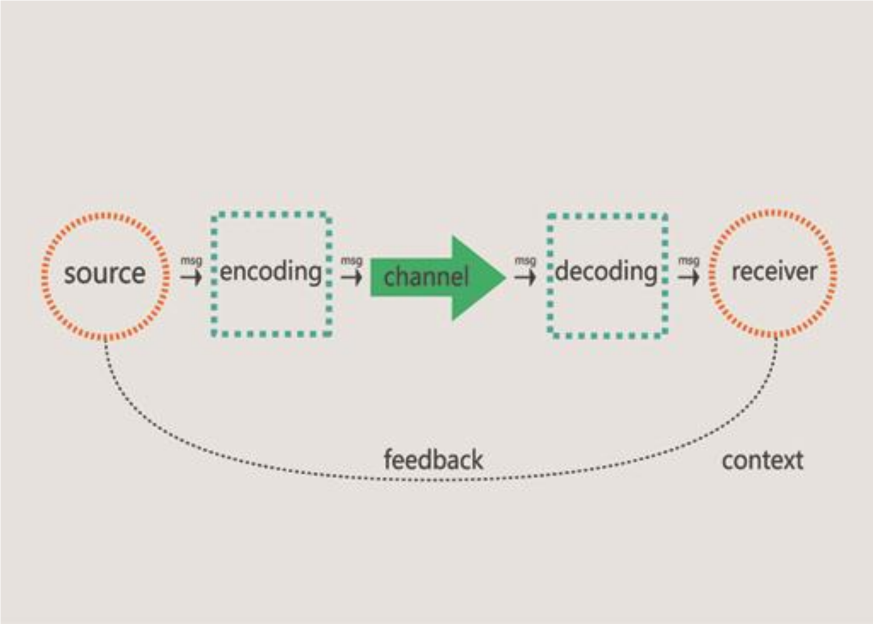
Notiune de comunicare

Schimb de informație între interlocutori

- Mesaj
- Emițător
- Codare
- Canal
- Decodare
- Receptor
- Raspuns
- Zgomot

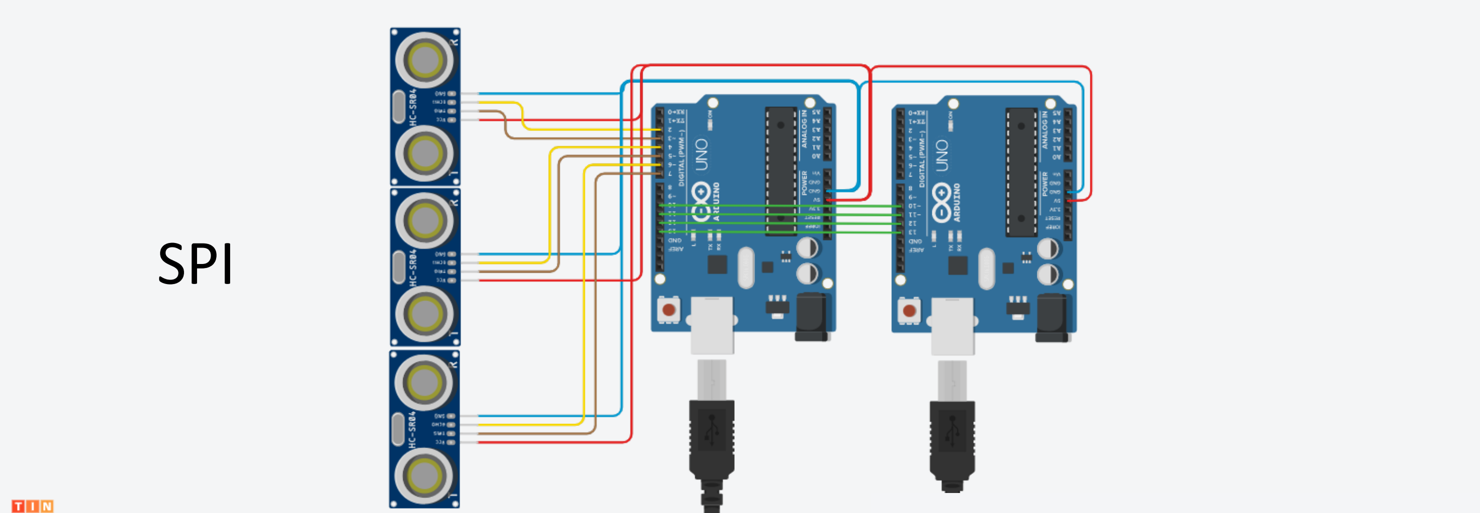


https://learntechit.com/the-process-of-communication/

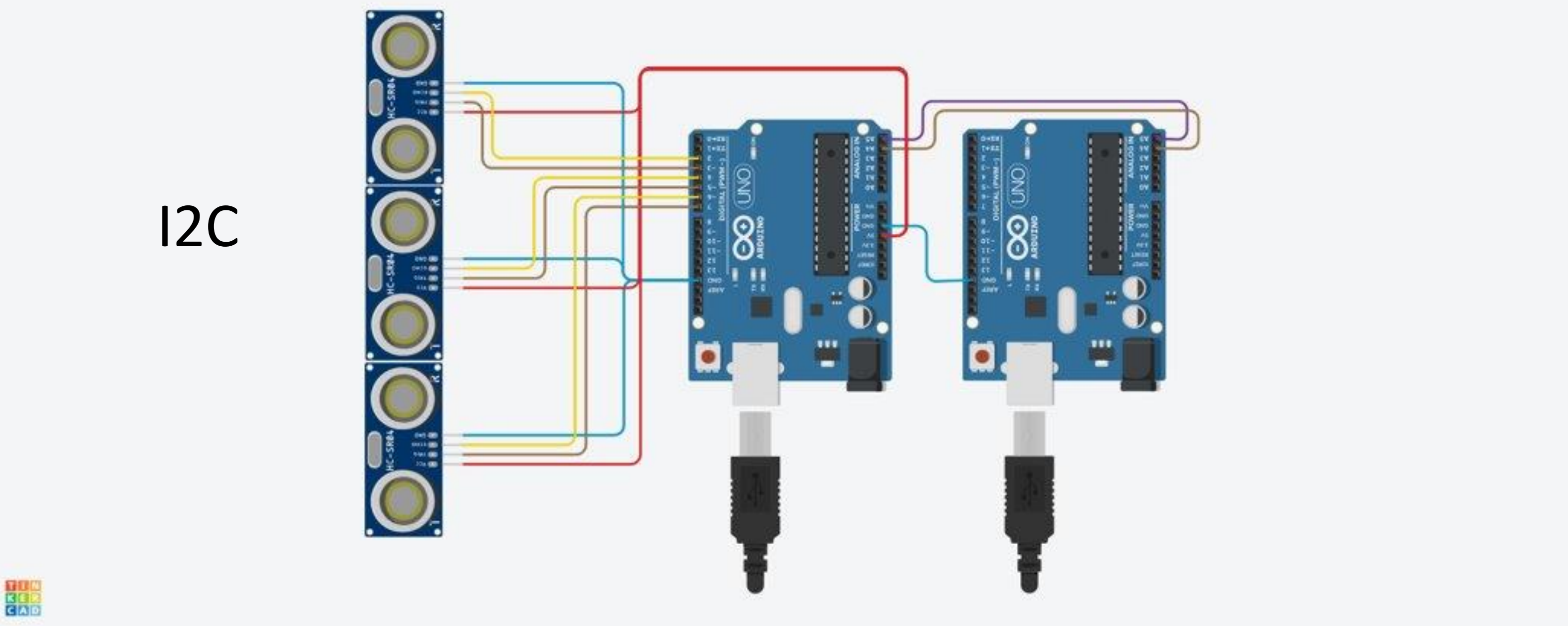


https://www.open.edu/openlearn/ocw/mod/oucontent/view.php?id=87012&section=4

Protocol Fizic - Digital Ultrasonic Sensor HCS-04



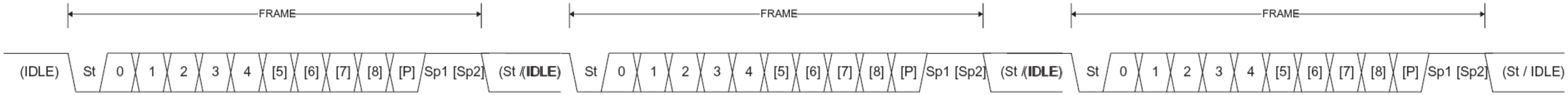
SPI



I2C

USART – Protocol Implementare

Protocol fizic



Protocol Logic

Protocol Frame								
STX	P nr	SRC	DST	P_ID	CMD	Payload (4 bytes)	SC	ETX

Protocol Frame								
STX	P nr	SRC	DST	P_ID	CMD	Payload (4 bytes)	SC	ETX

Protocol Frame								
STX	P nr	SRC	DST	P_ID	CMD	Payload (4 bytes)	SC	ETX

Protocol Frame								
STX	P nr	SRC	DST	P_ID	CMD	Payload (4 bytes)	SC	ETX

- Stx – 0x02
- Etx - 0x03
- Pnr – contorizare pachete
- SRC – ID emitator
- DST – ID receptor
- P_ID – tipul pachetului
- CMD – comada
- Payload – date pachet
- SC – suma de control

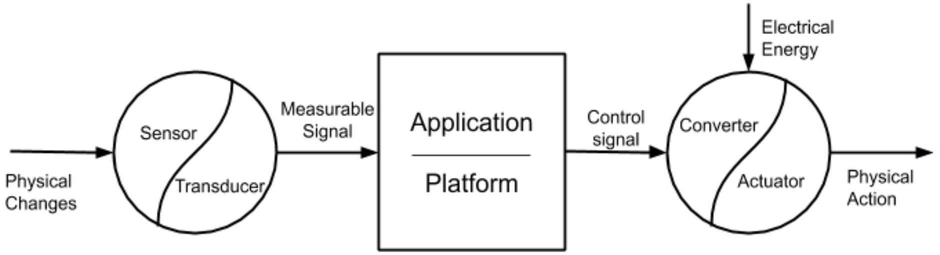
Emitere

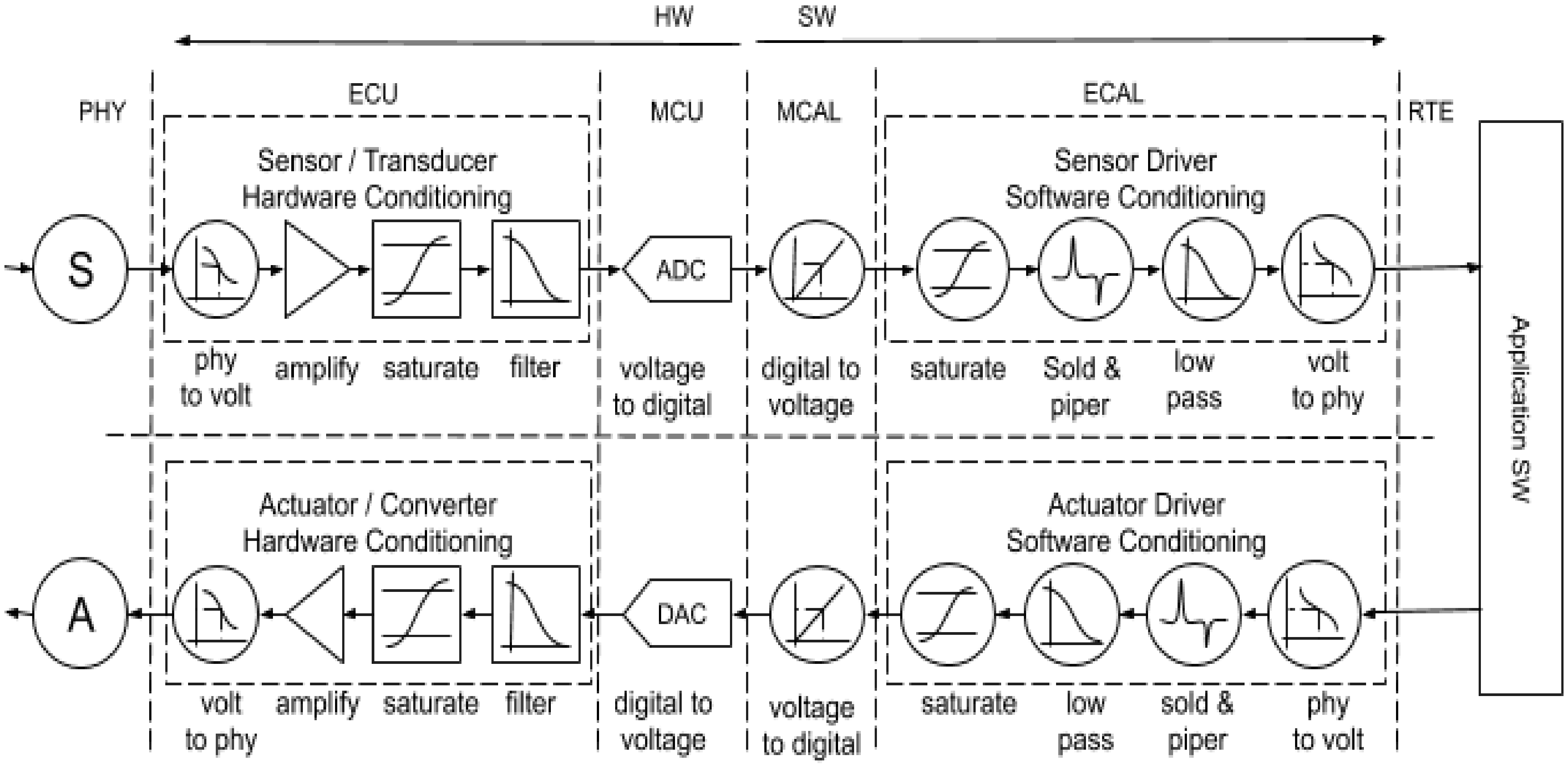
1. Selectie Date
2. Impachetare
3. Creare SC
4. Trimitere

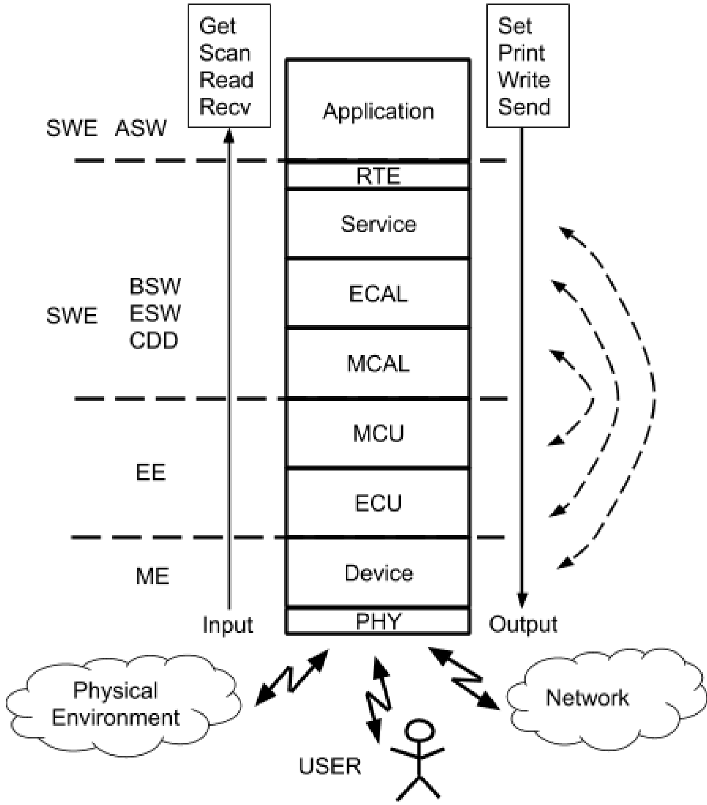
Receptie

1. Colectare byte
2. Buferizare
3. Verificare
4. Interpretare Date

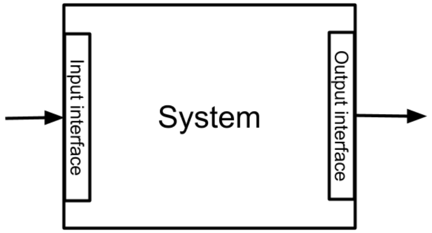
Comunicatii cu semnale

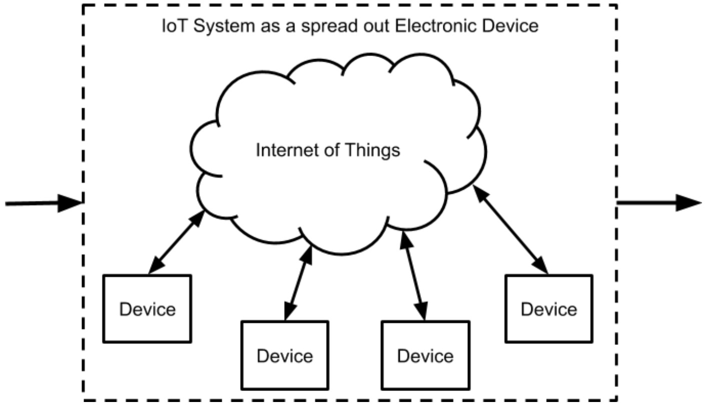


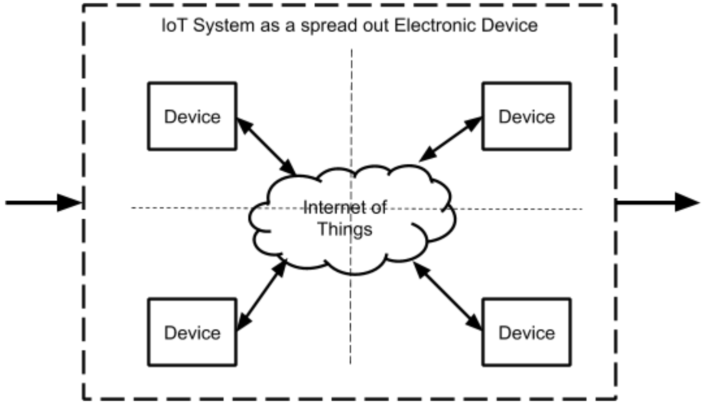




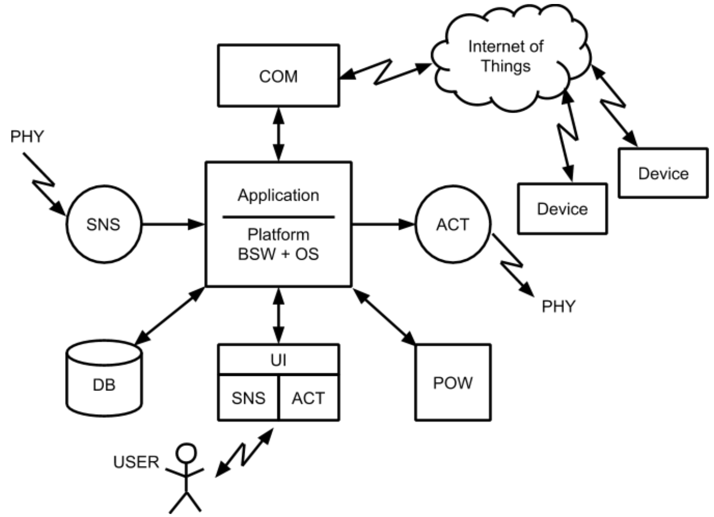
IoT Interactiuni



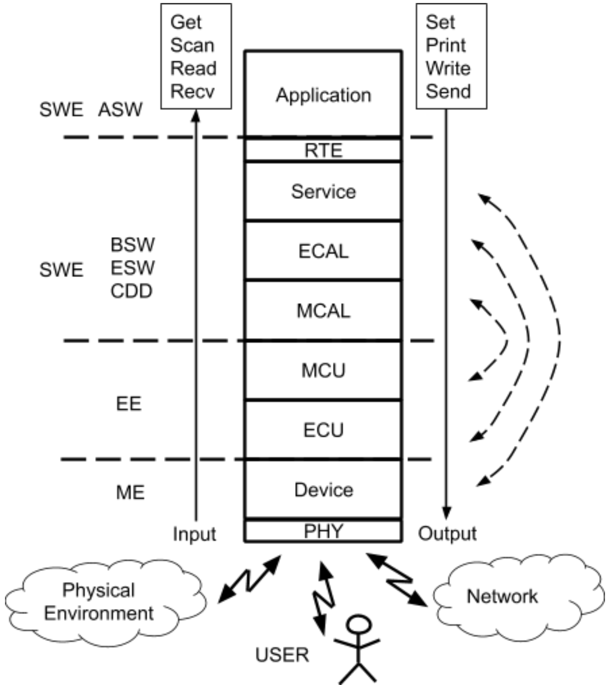




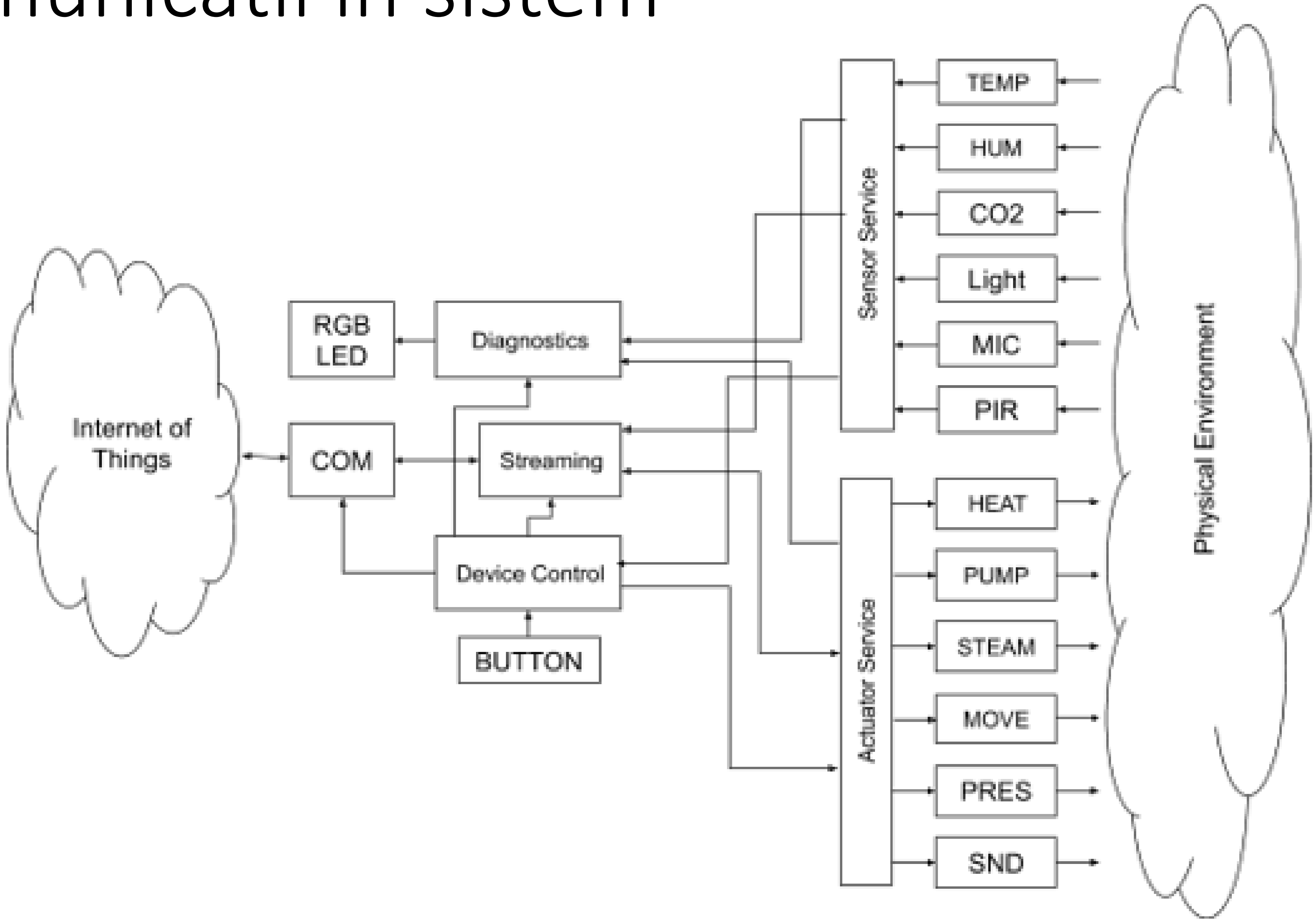
Device - Network



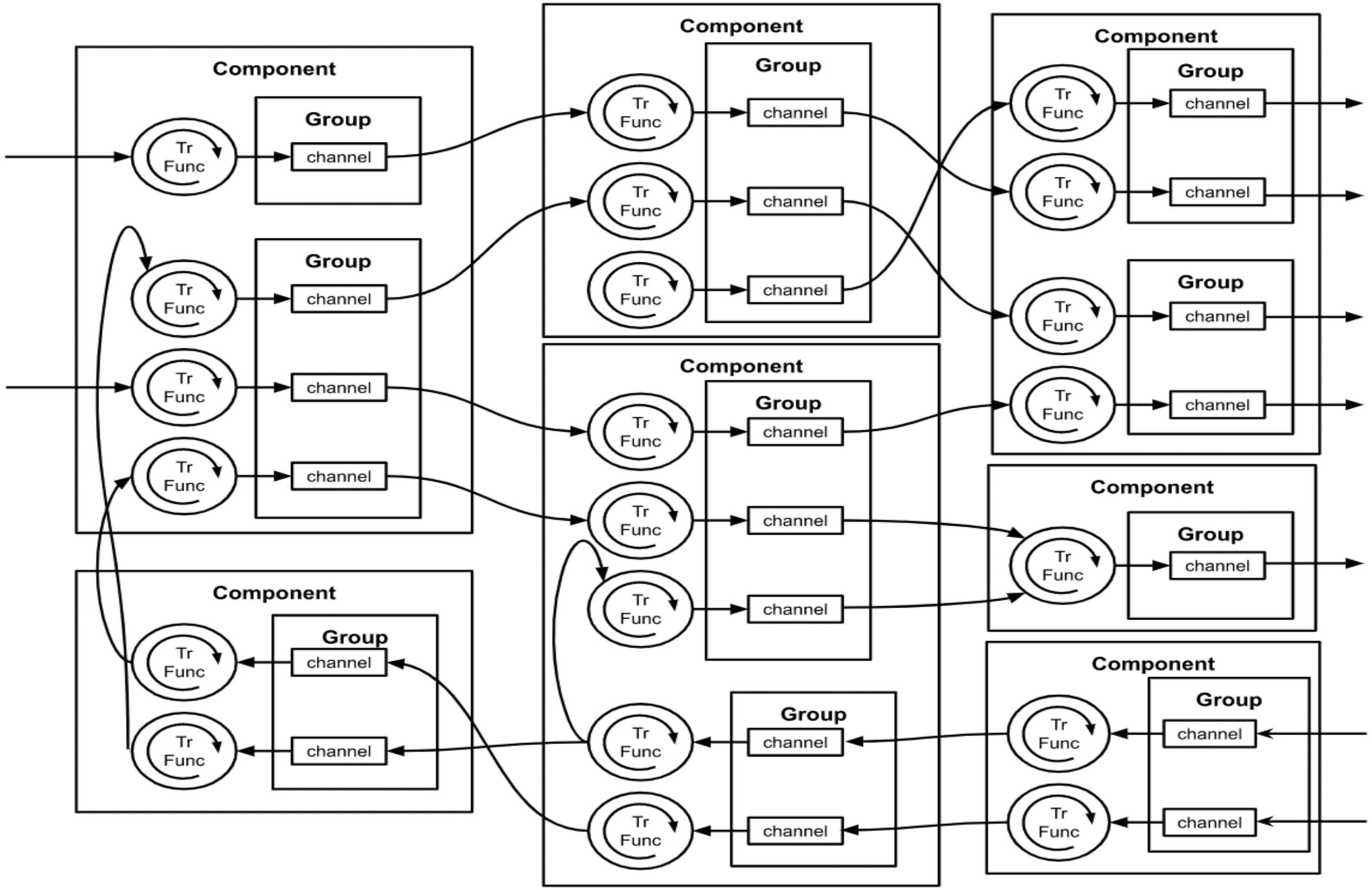
Device - Periferii

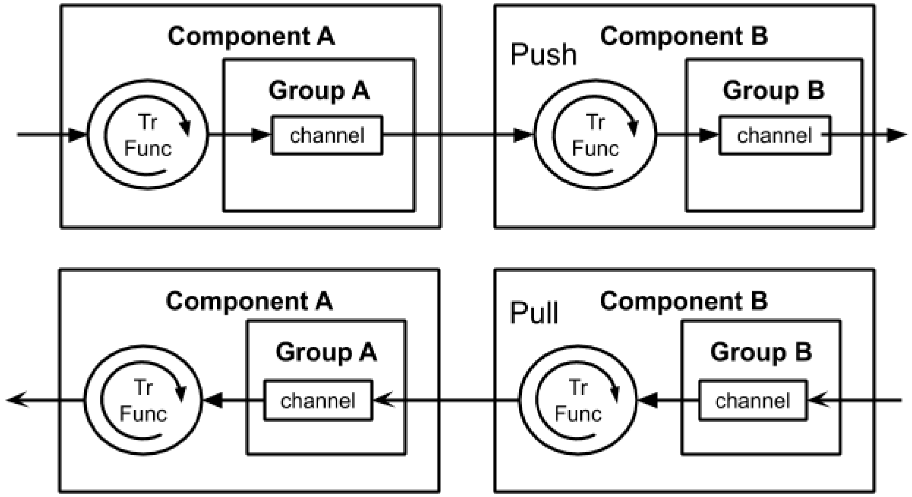


Comunicatii in sistem

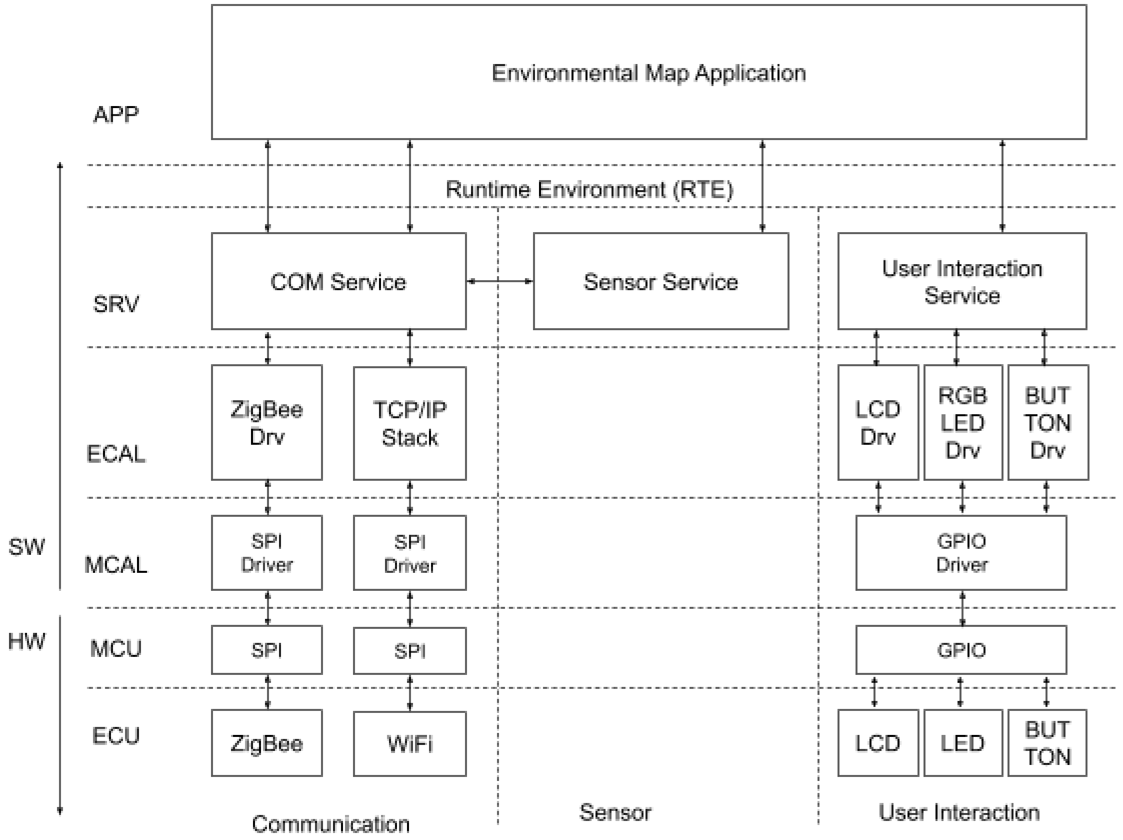


Vedere aplatizata a arhitecturii sistemului





Comunicatii prin nivele de abstractie



Environmental Map Application

APP

Runtime Environment (RTE)

SRV: COM Service, Sensor Service, User Interaction Service

ECAL: ZigBee Drv, TCP/IP Stack, LCD Drv, RGB LED Drv, BUT TON Drv

MCAL: SPI Driver, SPI Driver, GPIO Driver

MCU: SPI, SPI, GPIO

ECU: ZigBee, WiFi, LCD, LED, BUT TON

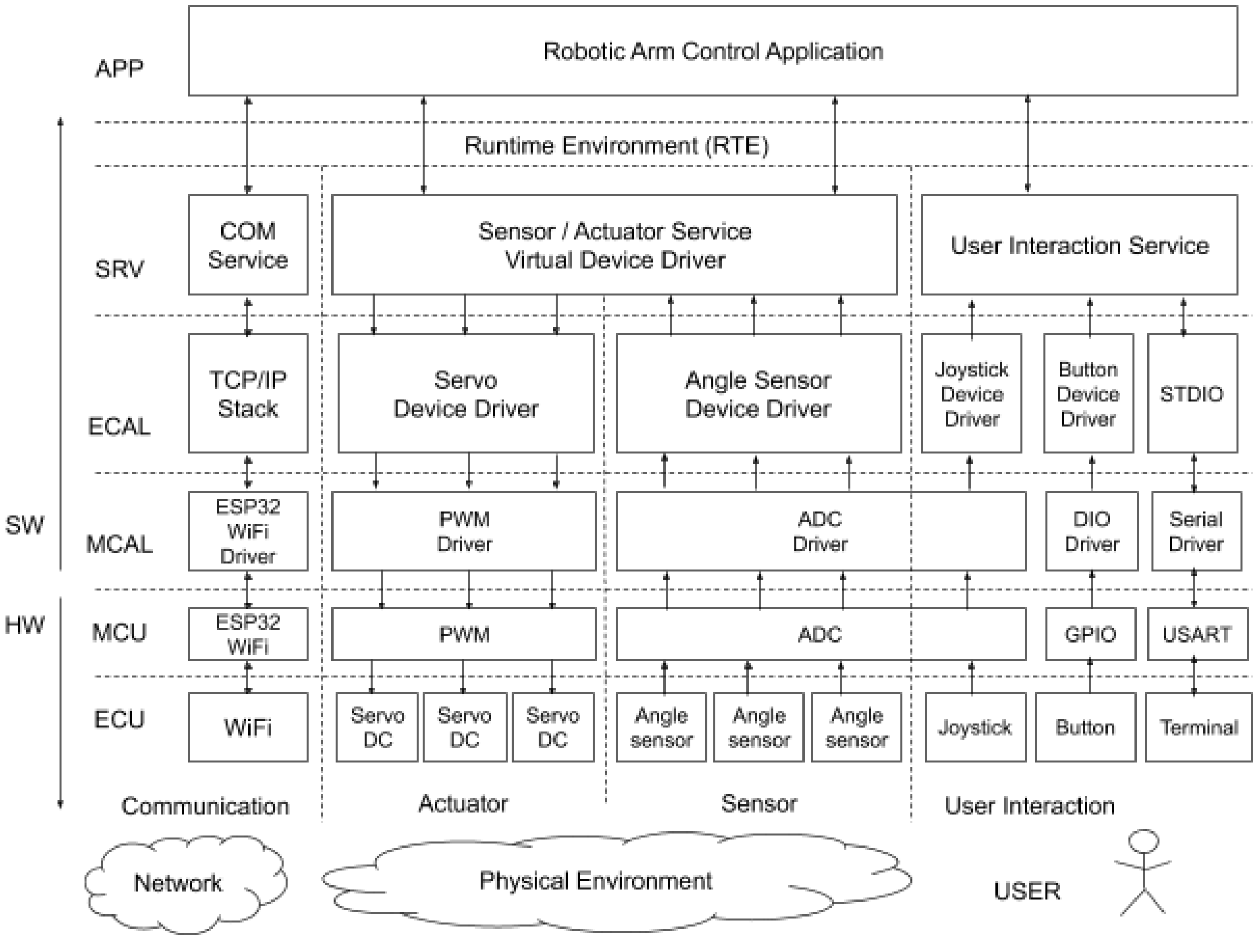
SW

HW

Communication

Sensor

User Interaction



Robotic Arm Control Application

APP

Runtime Environment (RTE)

SRV: COM Service, Sensor / Actuator Service Virtual Device Driver, User Interaction Service

ECAL: TCP/IP Stack, Servo Device Driver, Angle Sensor Device Driver, Joystick Device Driver, Button Device Driver, STDIO

MCAL: ESP32 WiFi Driver, PWM Driver, ADC Driver, DIO Driver, Serial Driver

MCU: ESP32 WiFi, PWM, ADC, GPIO, USART

ECU: WiFi, Servo DC, Servo DC, Servo DC, Angle sensor, Angle sensor, Angle sensor, Joystick, Button, Terminal

SW

HW

Communication

Actuator

Sensor

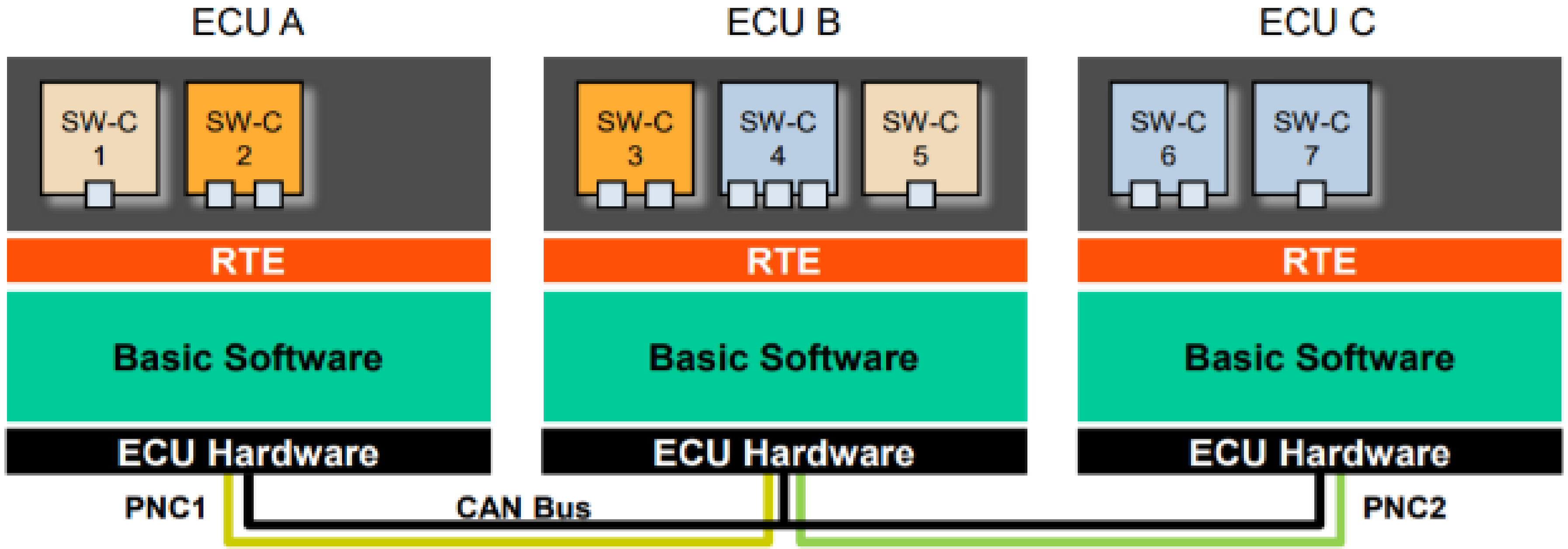
User Interaction

Network

Physical Environment

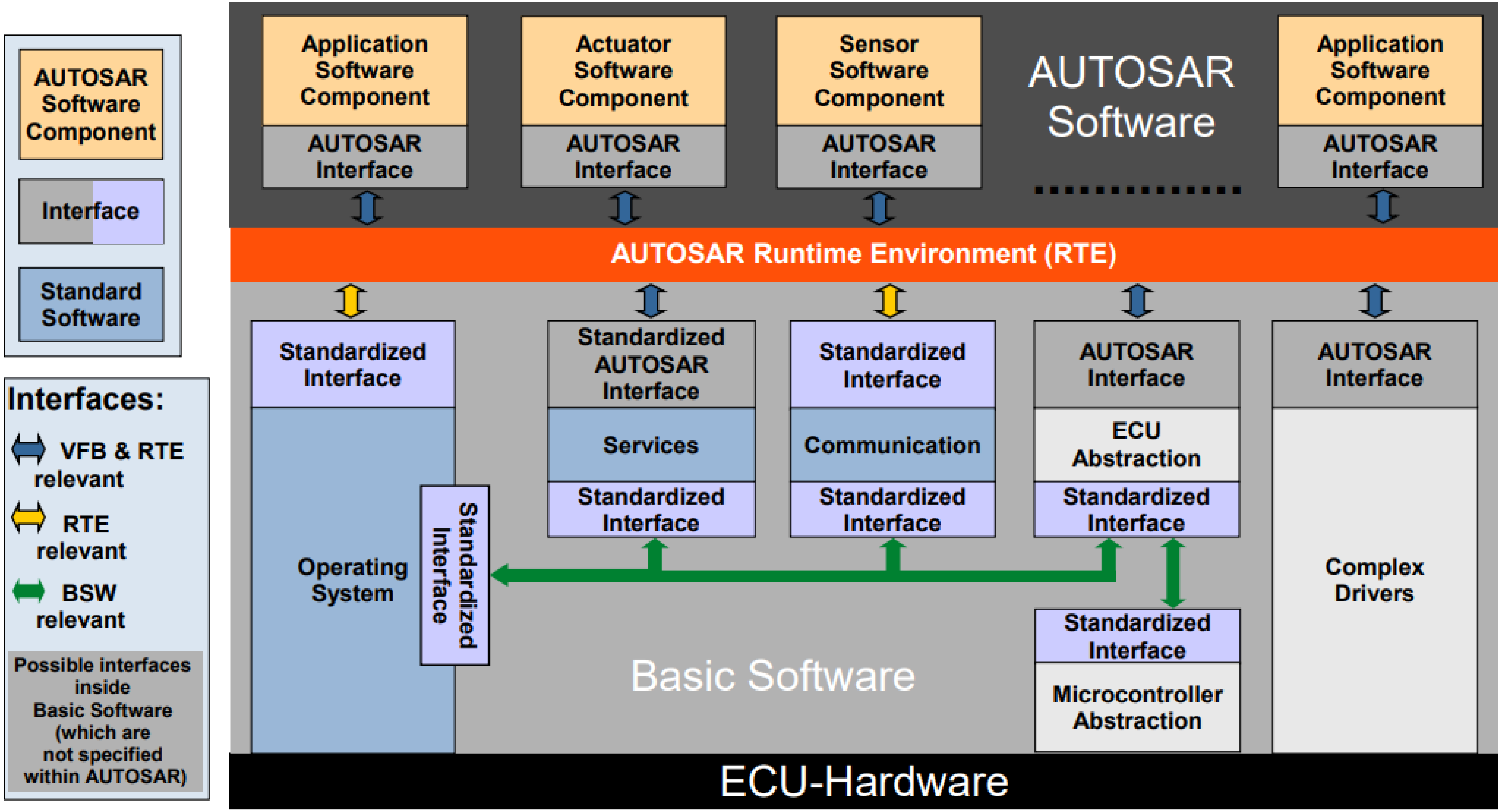
USER

Comunicatii intre componente



page id: 94ju5

Interfaces
Components and interfaces view (simplified)



AUTOSAR Software Component

Interface

Standard Software

Interfaces:

VFB & RTE relevant

RTE relevant

BSW relevant

Possible interfaces inside Basic Software (which are not specified within AUTOSAR)

Application Software Component

AUTOSAR Interface

Actuator Software Component

AUTOSAR Interface

Sensor Software Component

AUTOSAR Interface

AUTOSAR Software

....

Application Software Component

AUTOSAR Interface

AUTOSAR Runtime Environment (RTE)

Standardized Interface

Operating System

Standardized Interface

Standardized AUTOSAR Interface

Services

Standardized Interface

Standardized Interface

Communication

Standardized Interface

AUTOSAR Interface

ECU Abstraction

Standardized Interface

Standardized Interface

Microcontroller Abstraction

AUTOSAR Interface

Complex Drivers

Basic Software

ECU-Hardware

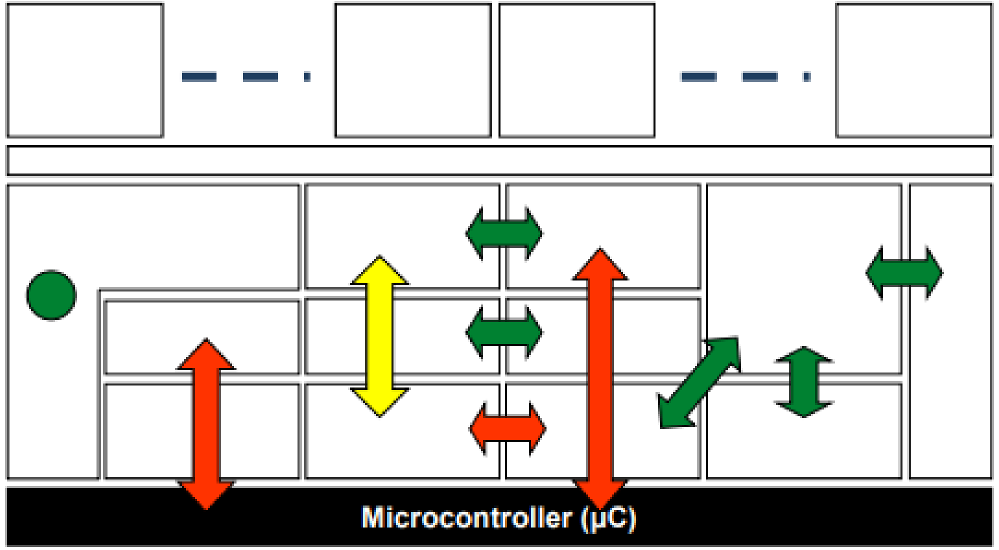
Note: This figure is incomplete with respect to the possible interactions between the layers.

https://www.autosar.org/fileadmin/user_upload/standards/classic/4-3/AUTOSAR_EXP_LayeredSoftwareArchitecture.pdf page id: a6ztr

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.



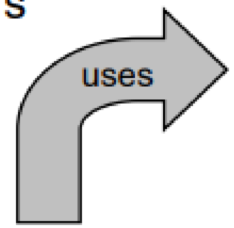
Vertical Interfaces

- ↕ 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.

page id: 1xdfr

Interfaces: General Rules
Layer Interaction Matrix

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



✓ allowed to use

× not allowed to use

Δ restricted use (callback only)

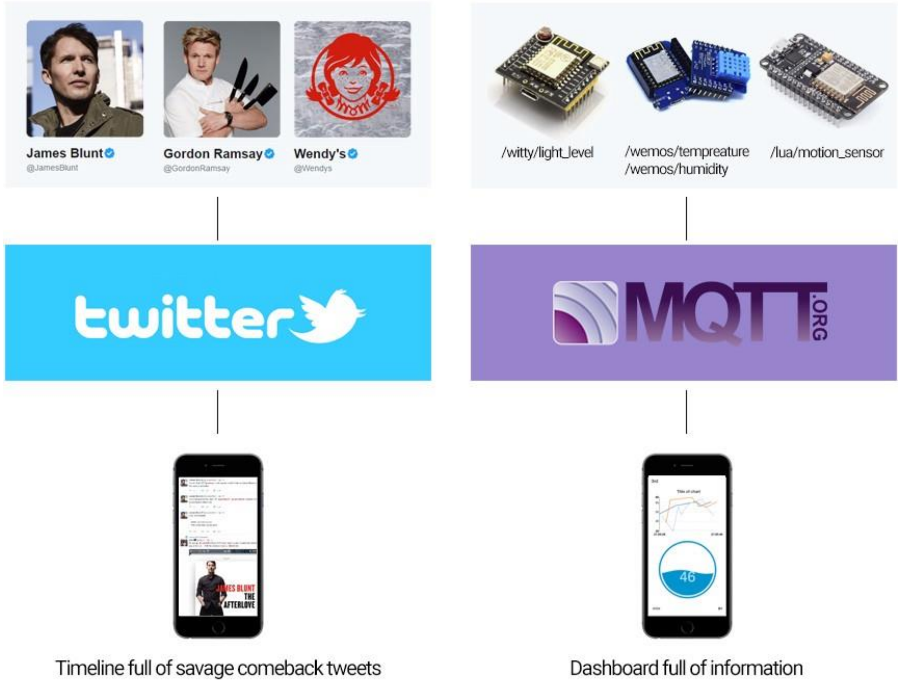
The matrix is read **row-wise**: ***Example:*** "I/O Drivers are allowed to use System Services and Hardware, but no other layers".

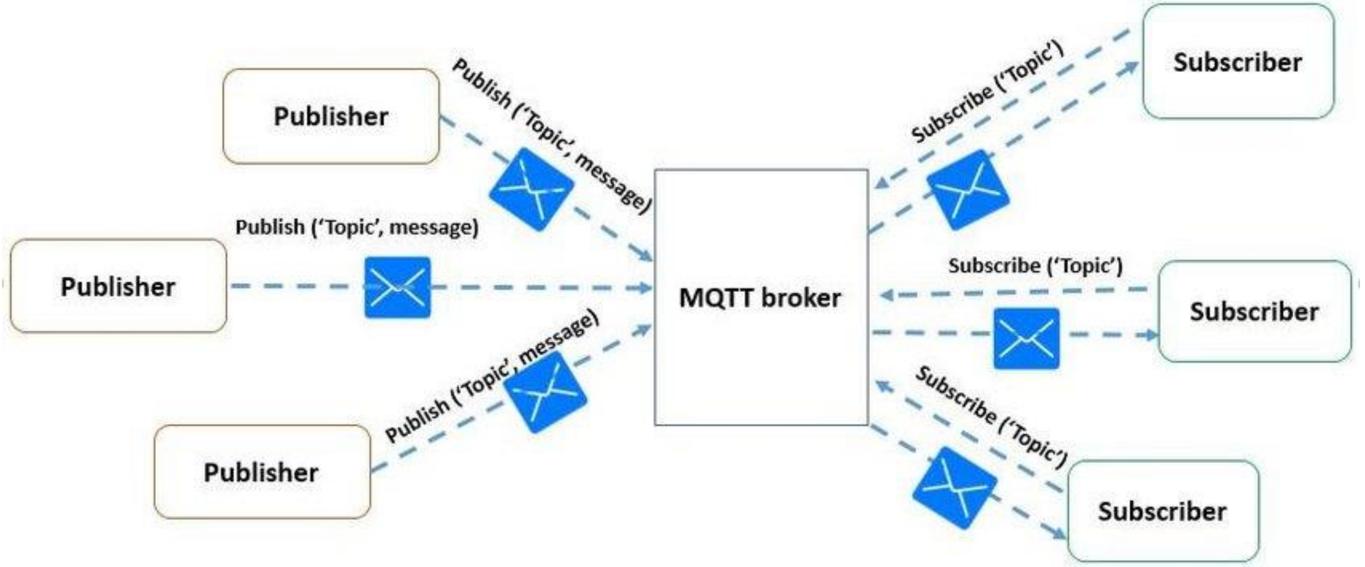
(gray background indicates "non-Basic Software" layers)

	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	✓

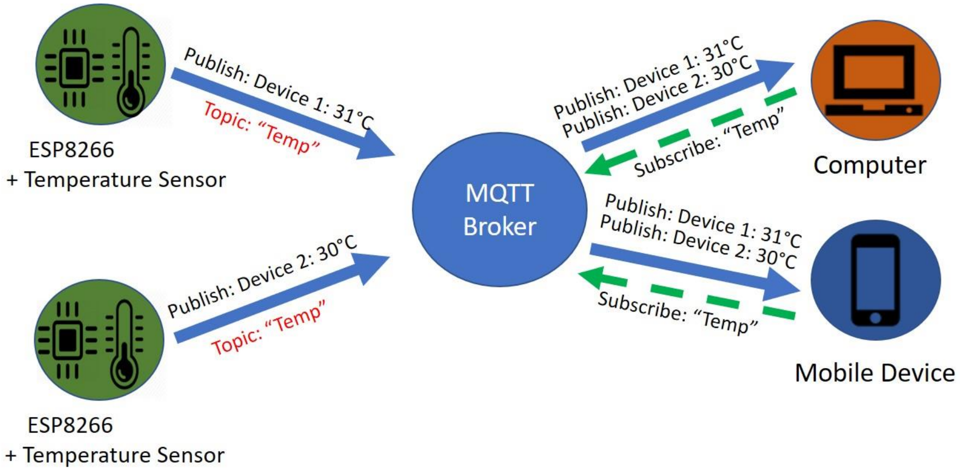
*: includes wired and wireless communication

IoT via MQTT

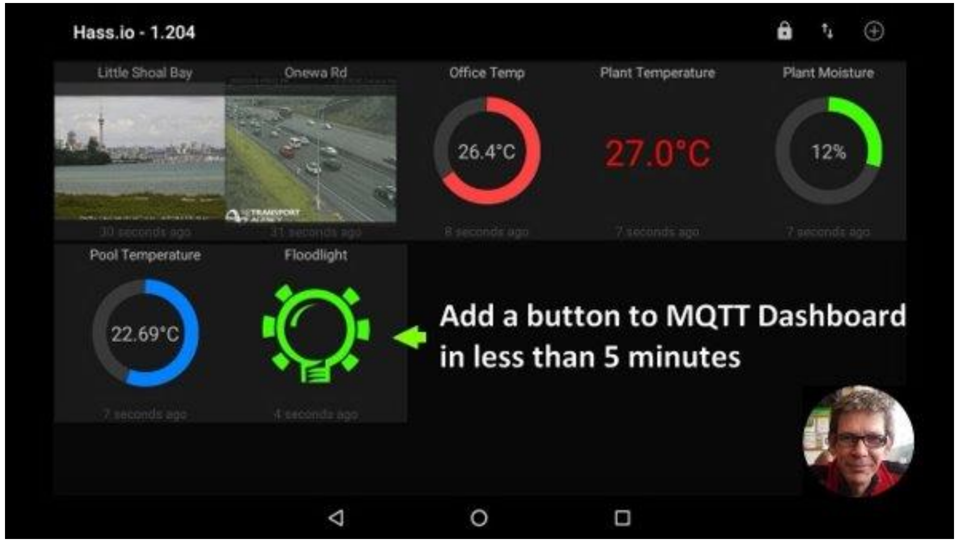

James Blunt
@JamesBlunt
Gordon Ramsay
@GordonRamsay
Wendy's
@Wendys
twitter
Timeline full of savage comeback tweets
/witty/light_level
/wemos/tempreature
/wemos/humidity
/lua/motion_sensor
MQTT.ORG
Dashboard full of information


Publisher
Publisher
Publisher
Publish ('Topic', message)
Publish ('Topic', message)
Publish ('Topic', message)
MQTT broker
Subscribe ('Topic')
Subscribe ('Topic')
Subscribe ('Topic')
Subscriber
Subscriber
Subscriber

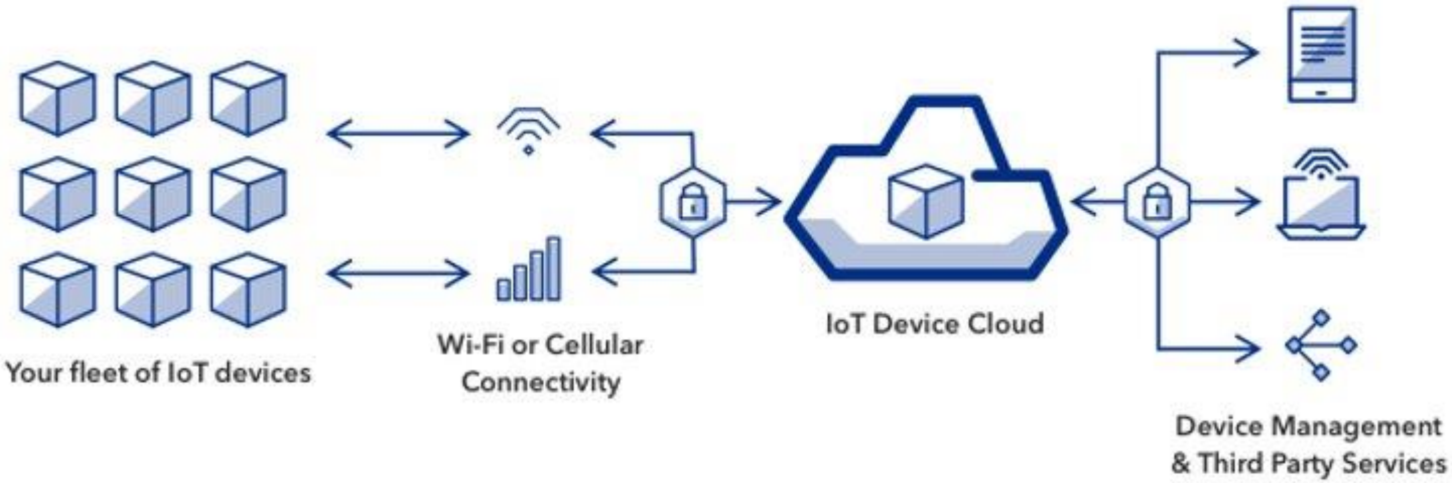
IoT via MQTT



Publish: Device 1: 31°C
Topic: "Temp"
ESP8266
+ Temperature Sensor
Publish: Device 2: 30°C
Topic: "Temp"
ESP8266
+ Temperature Sensor
MQTT Broker
Publish: Device 1: 31°C
Publish: Device 2: 30°C
Subscribe: "Temp"
Computer
Publish: Device 1: 31°C
Publish: Device 2: 30°C
Subscribe: "Temp"
Mobile Device

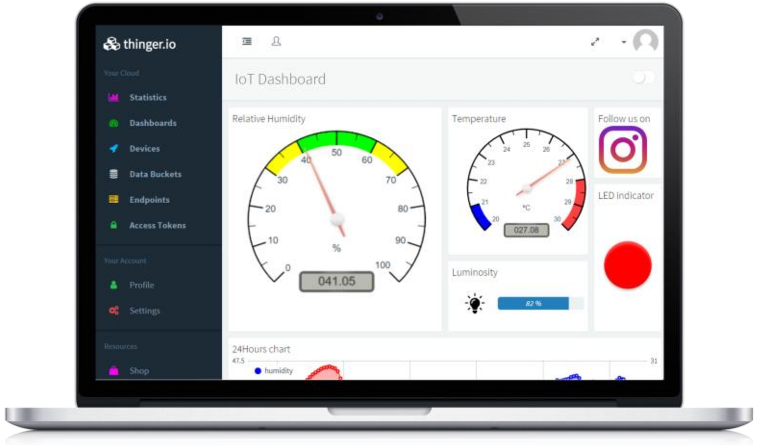


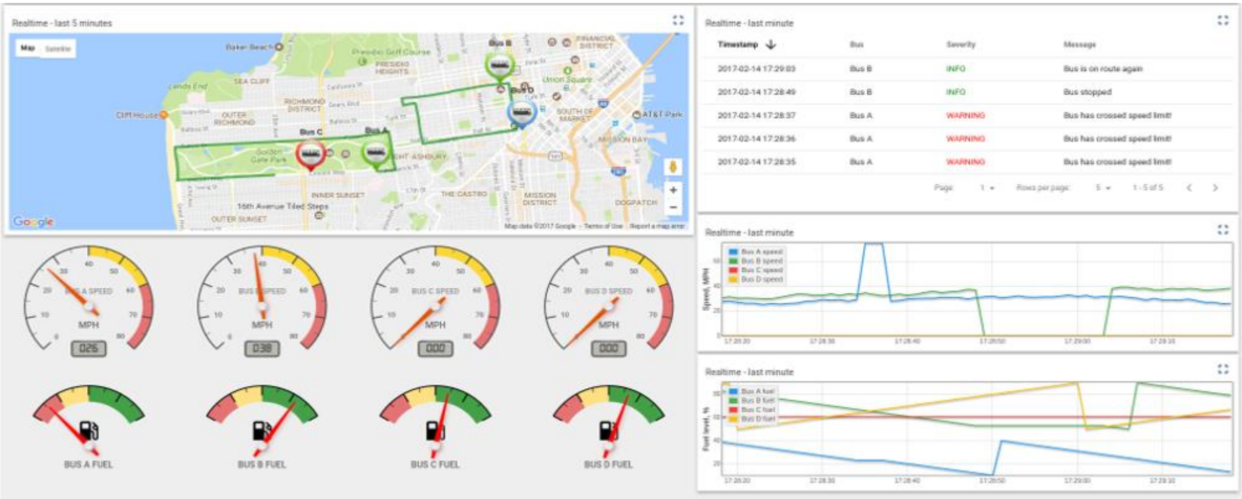
Hass.io - 1.204
Little Shoal Bay
Onewa Rd
Office Temp
26.4°C
Plant Temperature
27.0°C
Plant Moisture
12%
Pool Temperature
22.69°C
Floodlight
Add a button to MQTT Dashboard
in less than 5 minutes



Your fleet of IoT devices
Wi-Fi or Cellular Connectivity
IoT Device Cloud
Device Management & Third Party Services

MQTT – Monitorizare (dashboard)



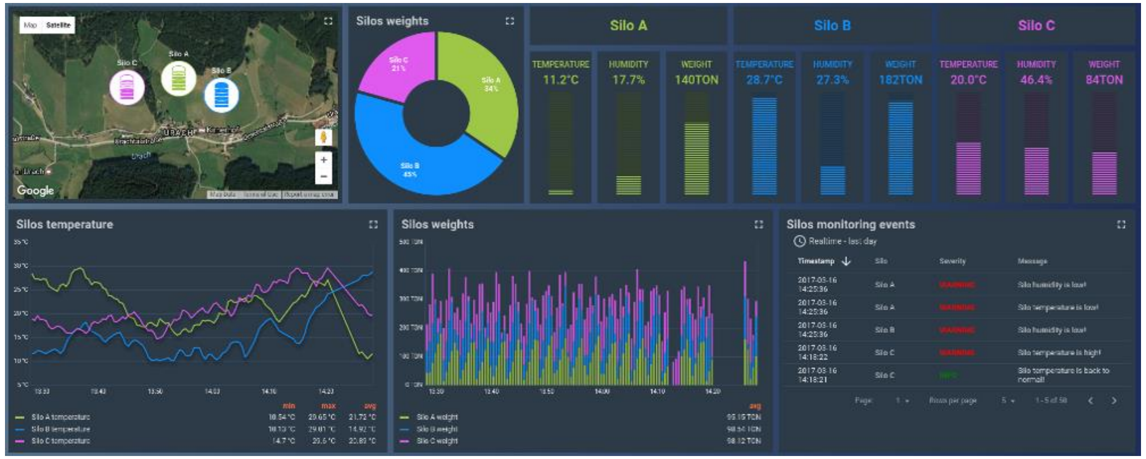


Realtime - last minute

Timestamp ↓	Bus	Severity	Message
2017-02-14 17:29:03	Bus B	INFO	Bus is on route again
2017-02-14 17:28:49	Bus B	INFO	Bus stopped
2017-02-14 17:28:37	Bus A	WARNING	Bus has crossed speed limit!
2017-02-14 17:28:36	Bus A	WARNING	Bus has crossed speed limit!
2017-02-14 17:28:35	Bus A	WARNING	Bus has crossed speed limit!

Page: 1 Rows per page: 5 1 - 5 of 5

https://thingsboard.io/



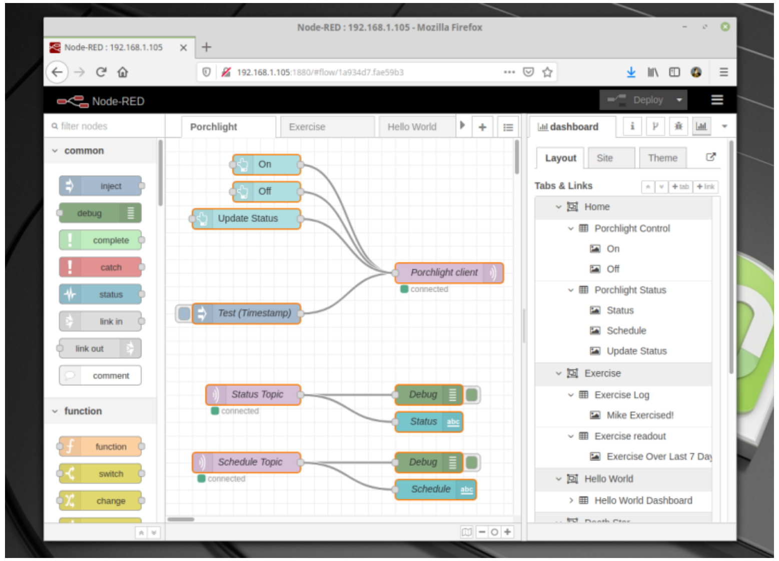
Silos monitoring events

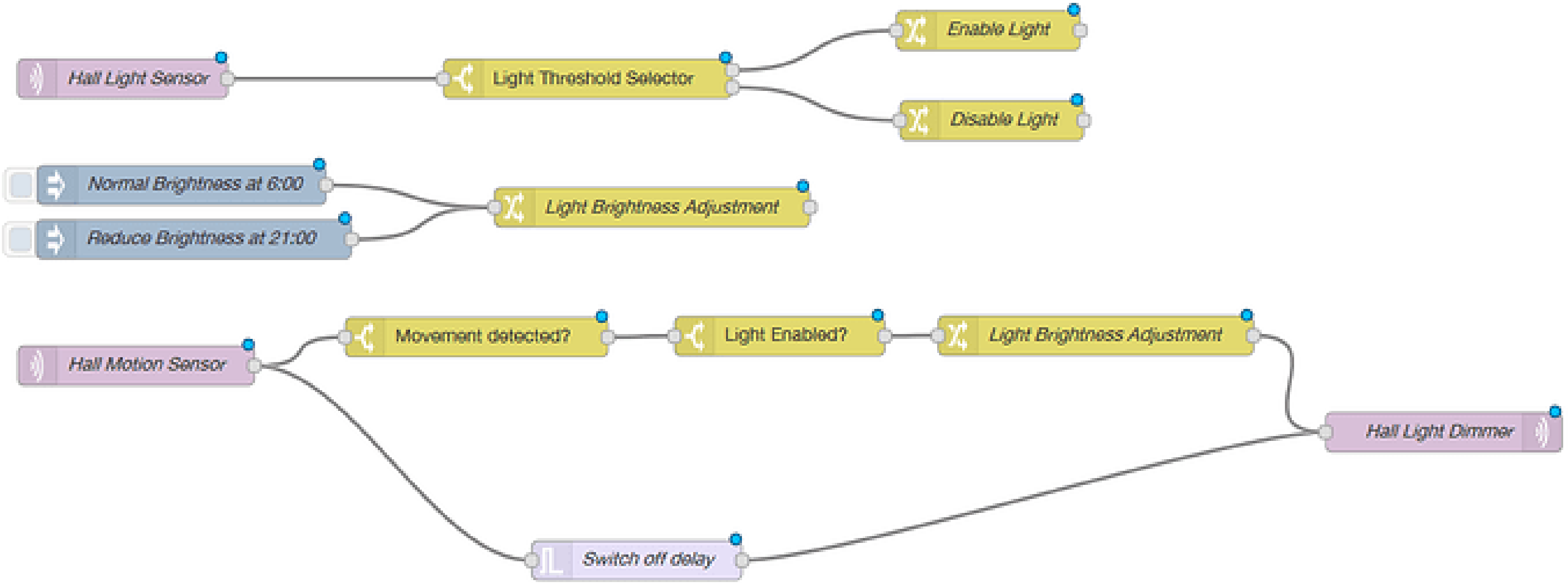
Realtime - last day

Timestamp ↓	Silo	Severity	Message
2017-03-16 14:25:36	Silo A	WARNING	Silo humidity is low!
2017-03-16 14:25:36	Silo A	WARNING	Silo temperature is low!
2017-03-16 14:25:36	Silo B	WARNING	Silo humidity is low!
2017-03-16 14:18:22	Silo C	WARNING	Silo temperature is high!
2017-03-16 14:18:21	Silo C	INFO	Silo temperature is back to normal!

Page: 1 Rows per page: 5 1 - 5 of 58

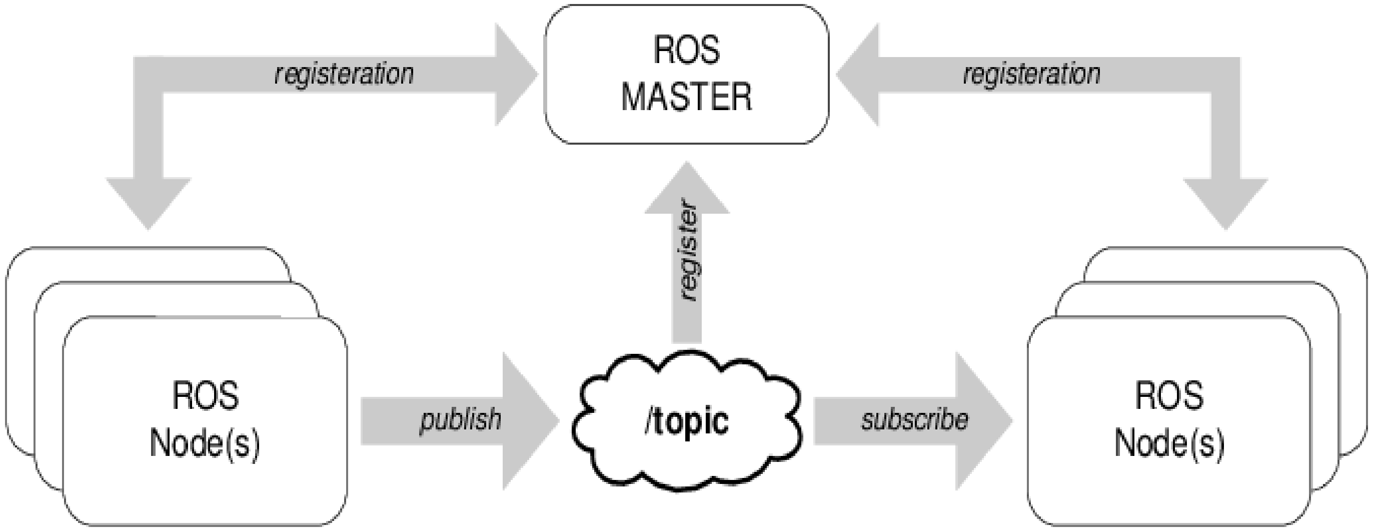
MQTT – Control (rule engine)





ROS – comunicare intre noduri

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 is an open-source project that extends the advanced capabilities of ROS to manufacturing automation and robotics. The ROS-Industrial repository includes interfaces for common industrial manipulators, grippers, sensors, and device networks. It also provides software libraries for automatic 2D/3D sensor calibration, process path/motion planning, applications like Scan-N-Plan, developer tools like the Qt Creator ROS Plugin, and training curriculum that is specific to the needs of manufacturers. ROS-I is supported by an international Consortium of industry and research members. ROS-Industrial:

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