Internet Of Things

… from electronics to the cloud

https://m2siame.univ-tlse3.fr

Dr. François Thiebolt / IRIT

thiebolt@irit.fr

Photo: Parallella board

Université Paul Sabatier TOULOUSE III

IRIT

neOCampus

Plan

General

- Part I – 1 Principles | end-devices, chip-level buses, …
- Part I – 2 SBC, communication chips, Industrial modules, …
- Part II – 1 IoT architecture & infrastructure protocols,
- Part II – 2 Data protocols, services and applications,
- Part III – The neOCampus use case.

Download these slides at https://m2siame.univ-tlse3.fr/teaching/francois/IoT_course.pdf

What's all this IoT stuff, Anyhow* ?

IoT is mainly Machine to Machine (M2M):
applications that communicate without human intervention.

- Habitat

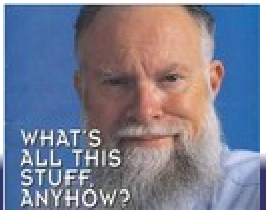
- Energy

- Transport

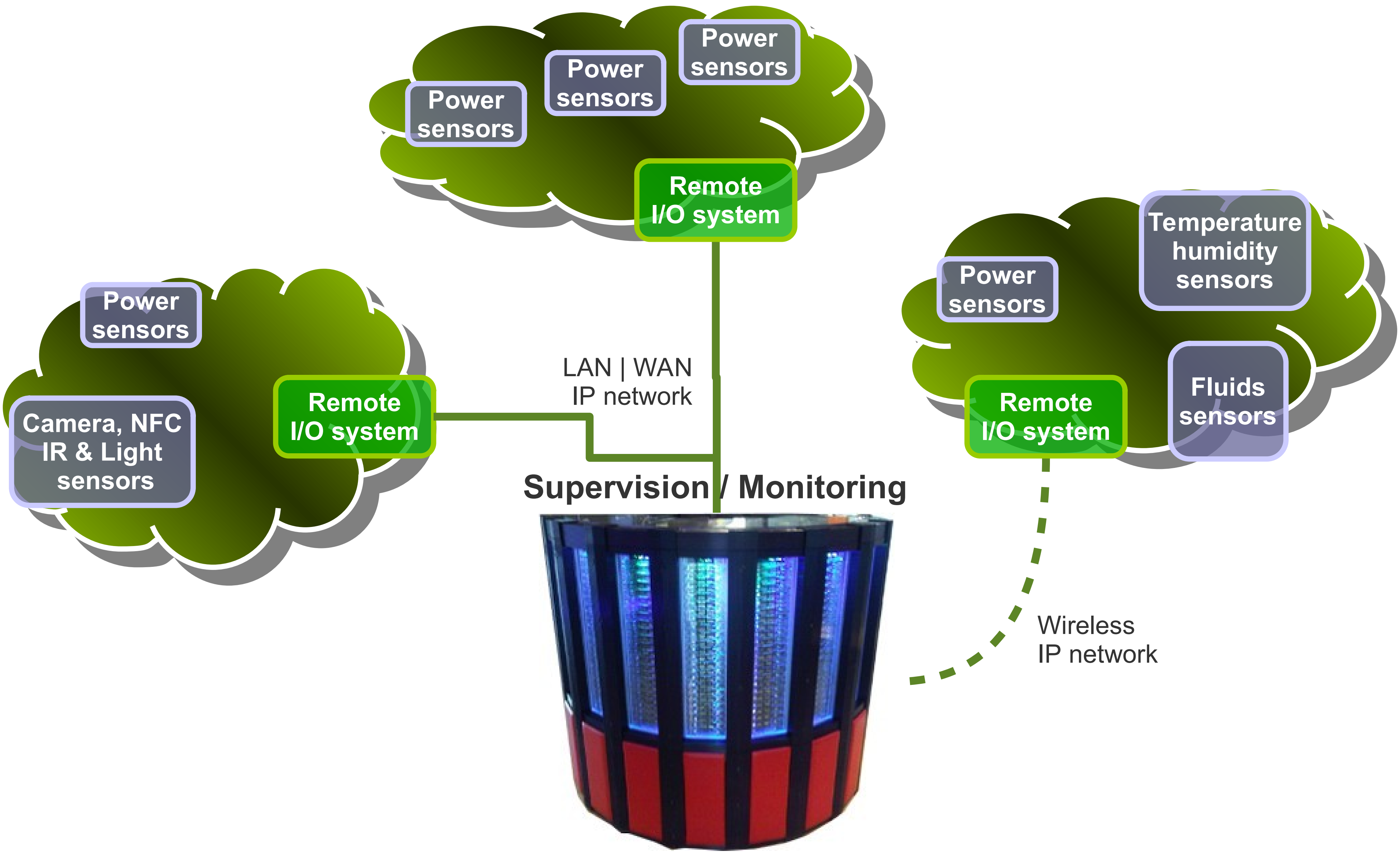
IoT devices everywhere ...

Connected device ≠ Intelligent device!

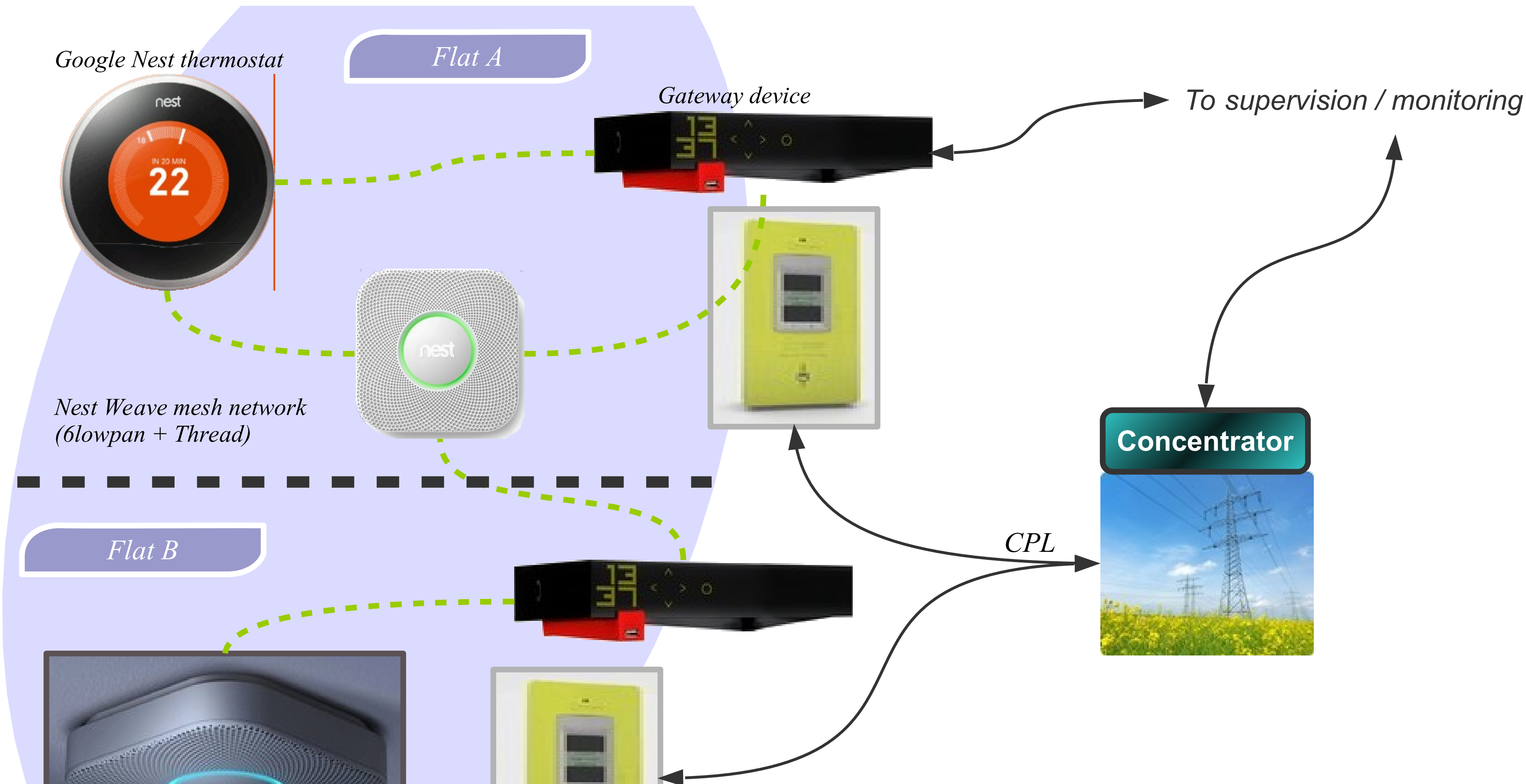
** Remember Ti's great analog engineer, Bob Pease*



Overview

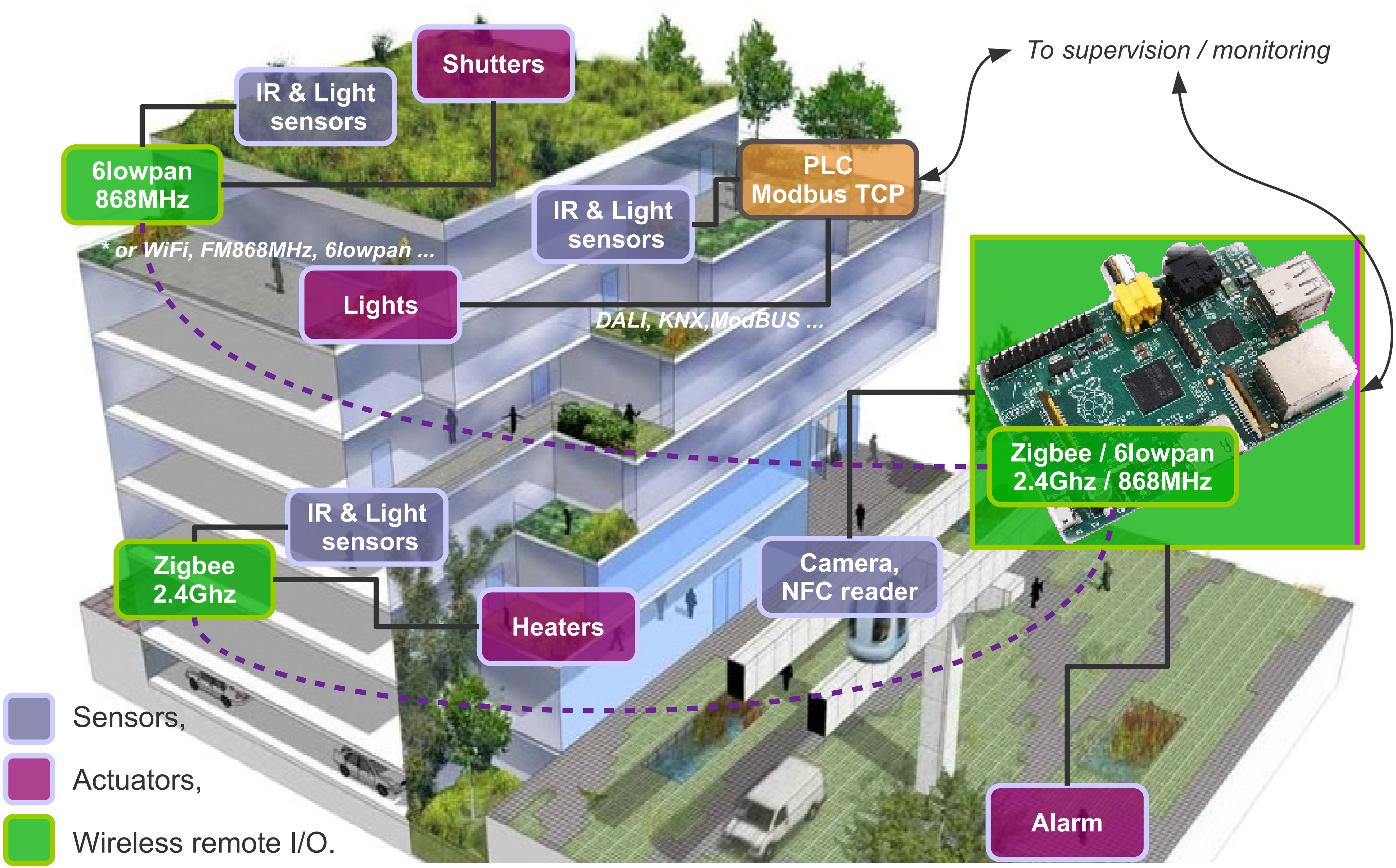

Power sensors
Power sensors
Power sensors
Remote I/O system
Power sensors
Camera, NFC IR & Light sensors
Remote I/O system
LAN | WAN IP network
Power sensors
Temperature humidity sensors
Remote I/O system
Fluids sensors
Supervision / Monitoring
Wireless IP network

Overview

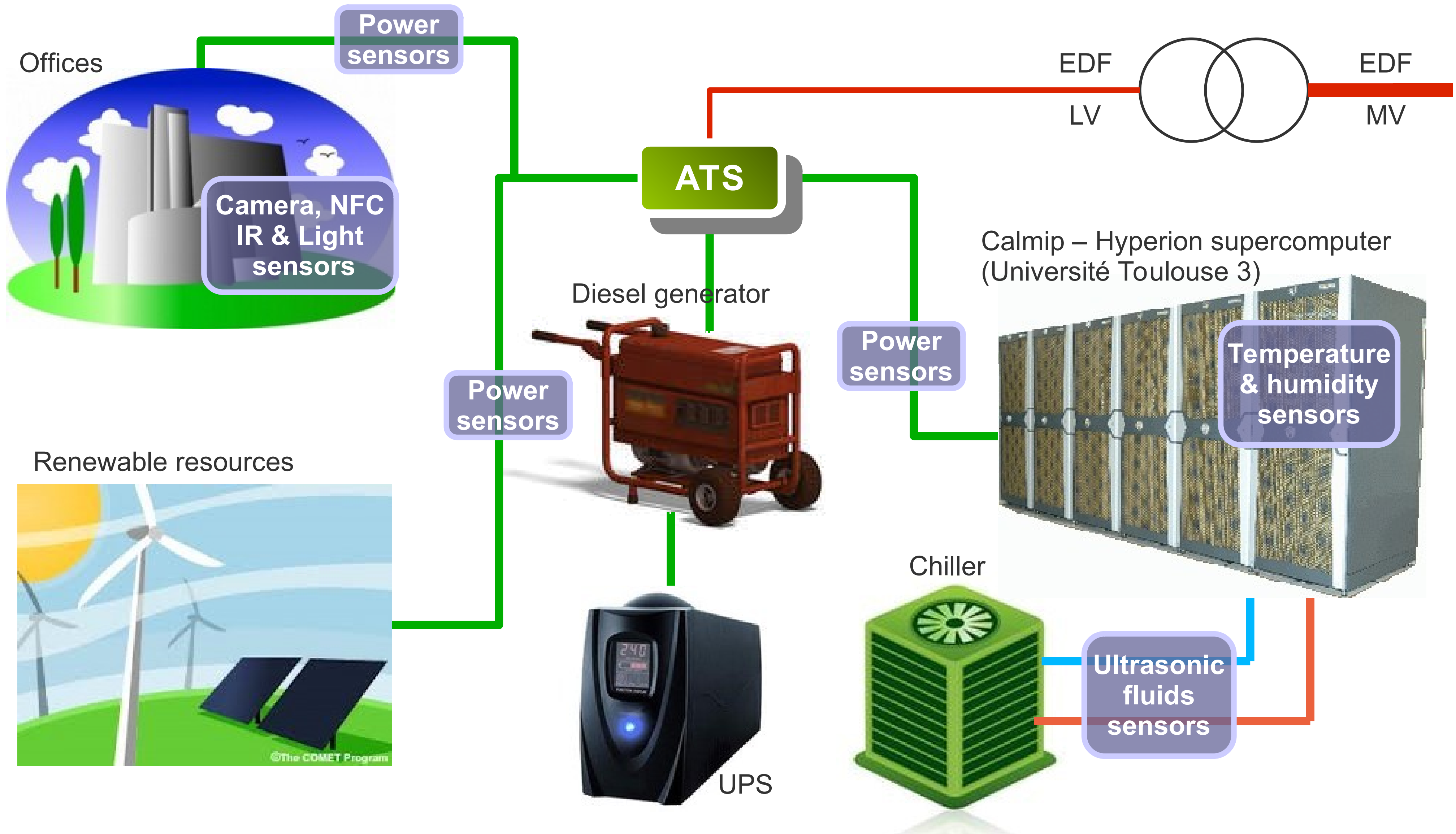


RT2012
Mettre en œuvre des systèmes permettant de mesurer ou d'estimer la consommation d'énergie de chaque logement, <u>par type d'énergie.</u>

Overview

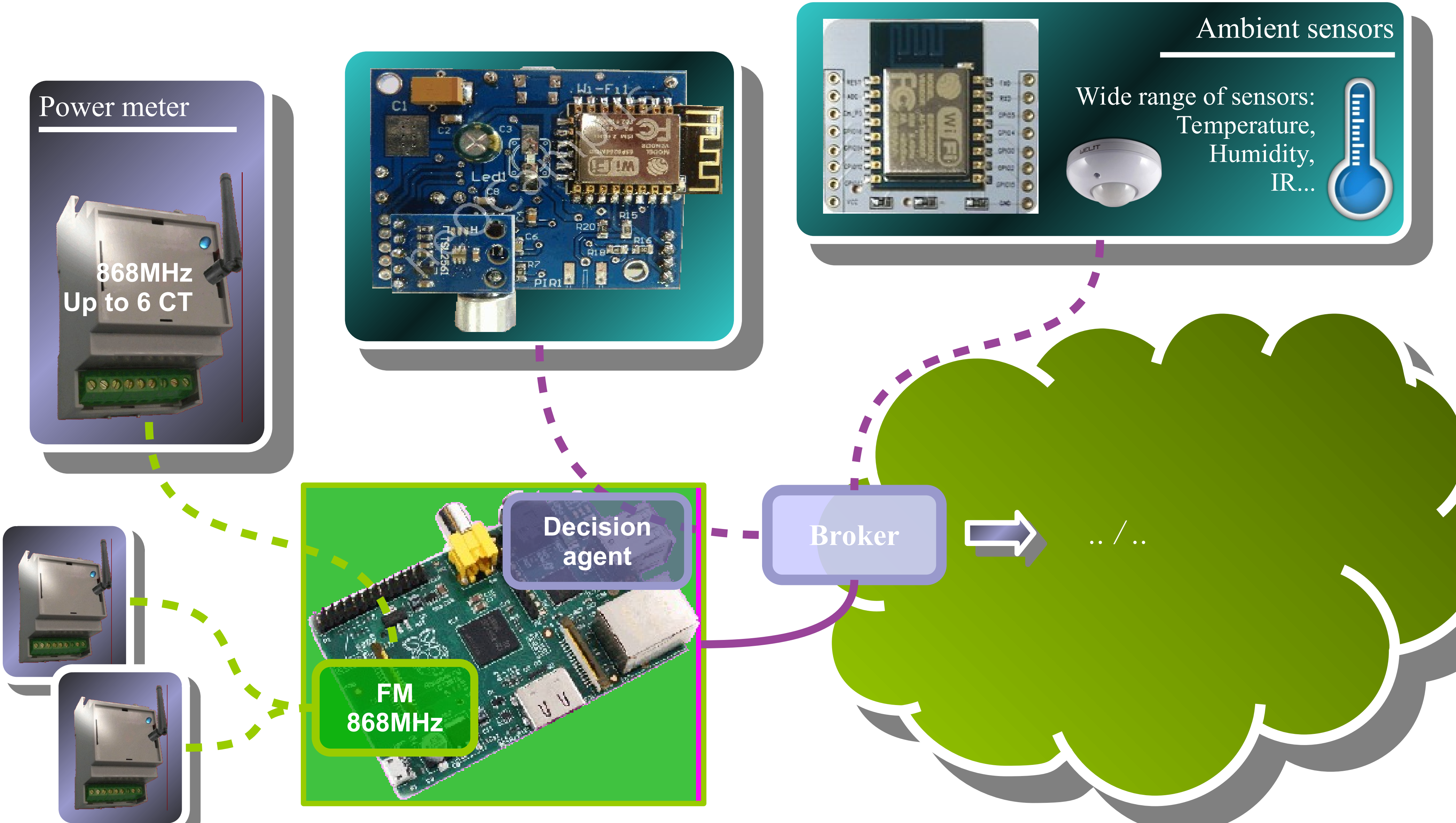

Shutters
IR & Light sensors
6lowpan 868MHz
PLC Modbus TCP
To supervision / monitoring
IR & Light sensors
* or WiFi, FM868MHz, 6lowpan ...
Lights
DALI, KNX,ModBUS ...
Zigbee / 6lowpan 2.4Ghz / 868MHz
IR & Light sensors
Zigbee 2.4Ghz
Camera, NFC reader
Heaters
Alarm
Sensors,
Actuators,
Wireless remote I/O.

Overview



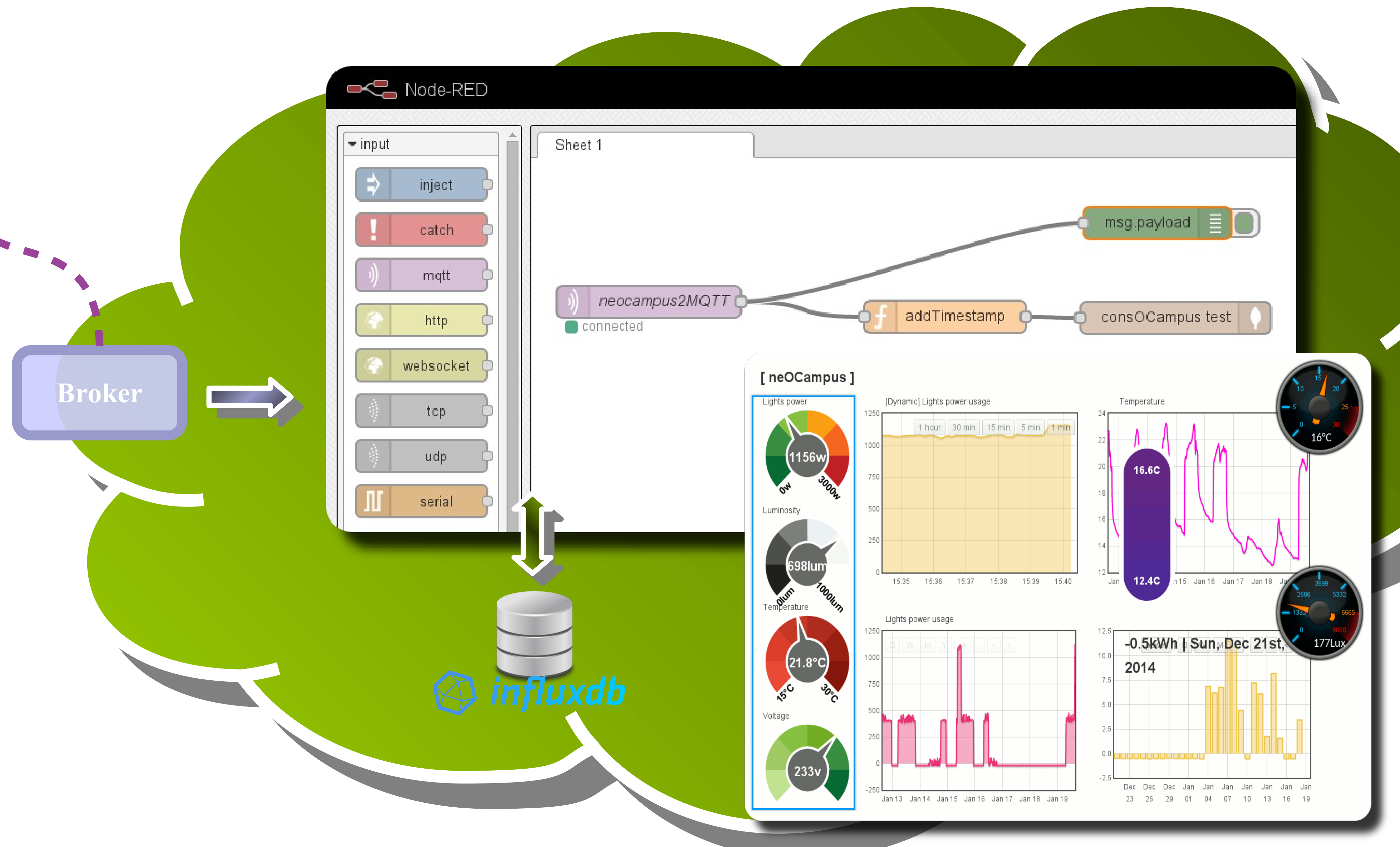
A microgrid is a localized grouping of electricity generation, energy storage, and loads that normally operates connected to a traditional centralized grid (macrogrid). This single point of common coupling with the macrogrid can be disconnected. The microgrid can then function autonomously. [Wikipedia]

Overview | neOCampus



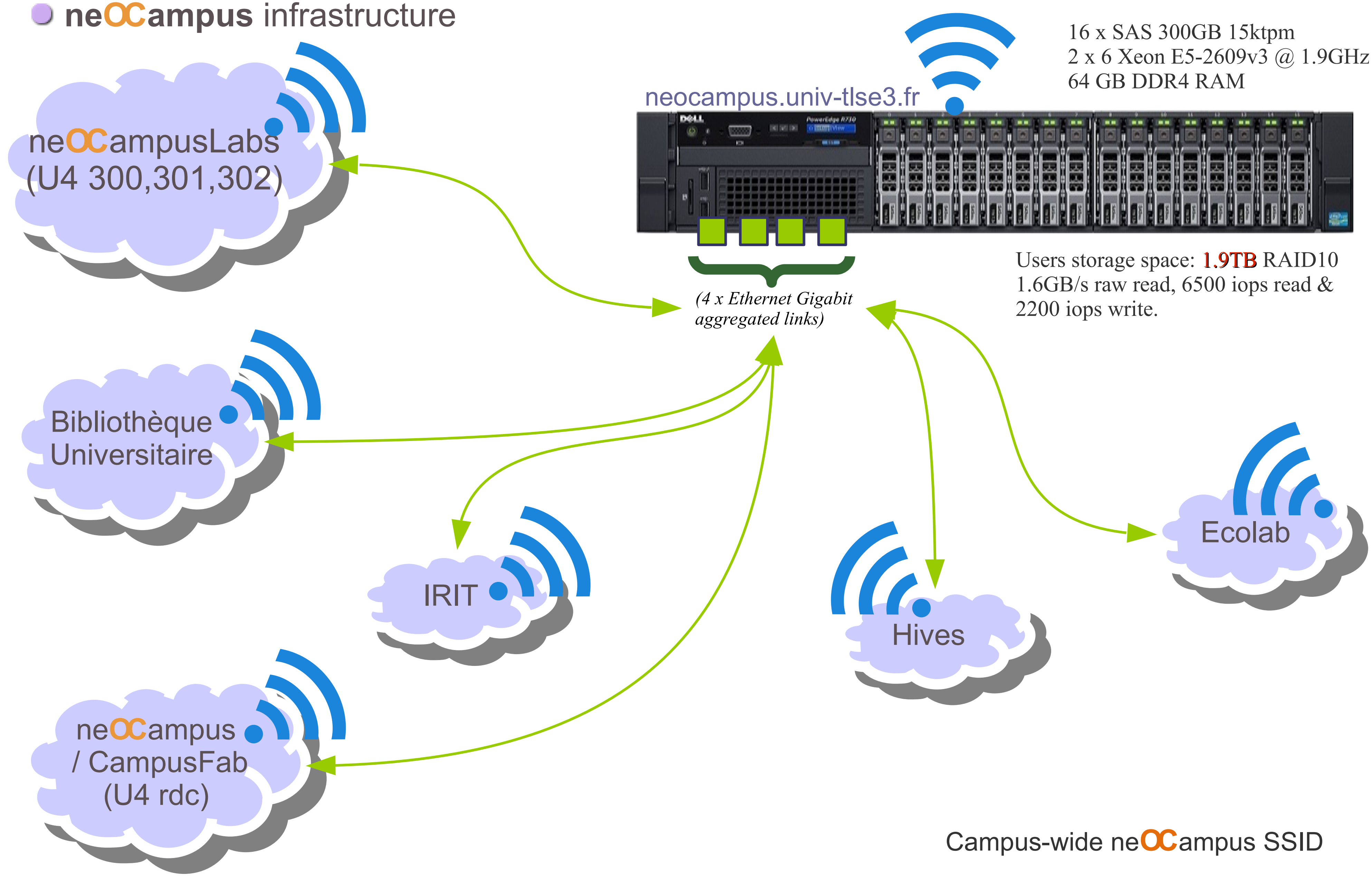
with its 868MHz transceiver, RPi acts as a gateway for those 868MHz enabled power-meters.

Overview | neOCampus



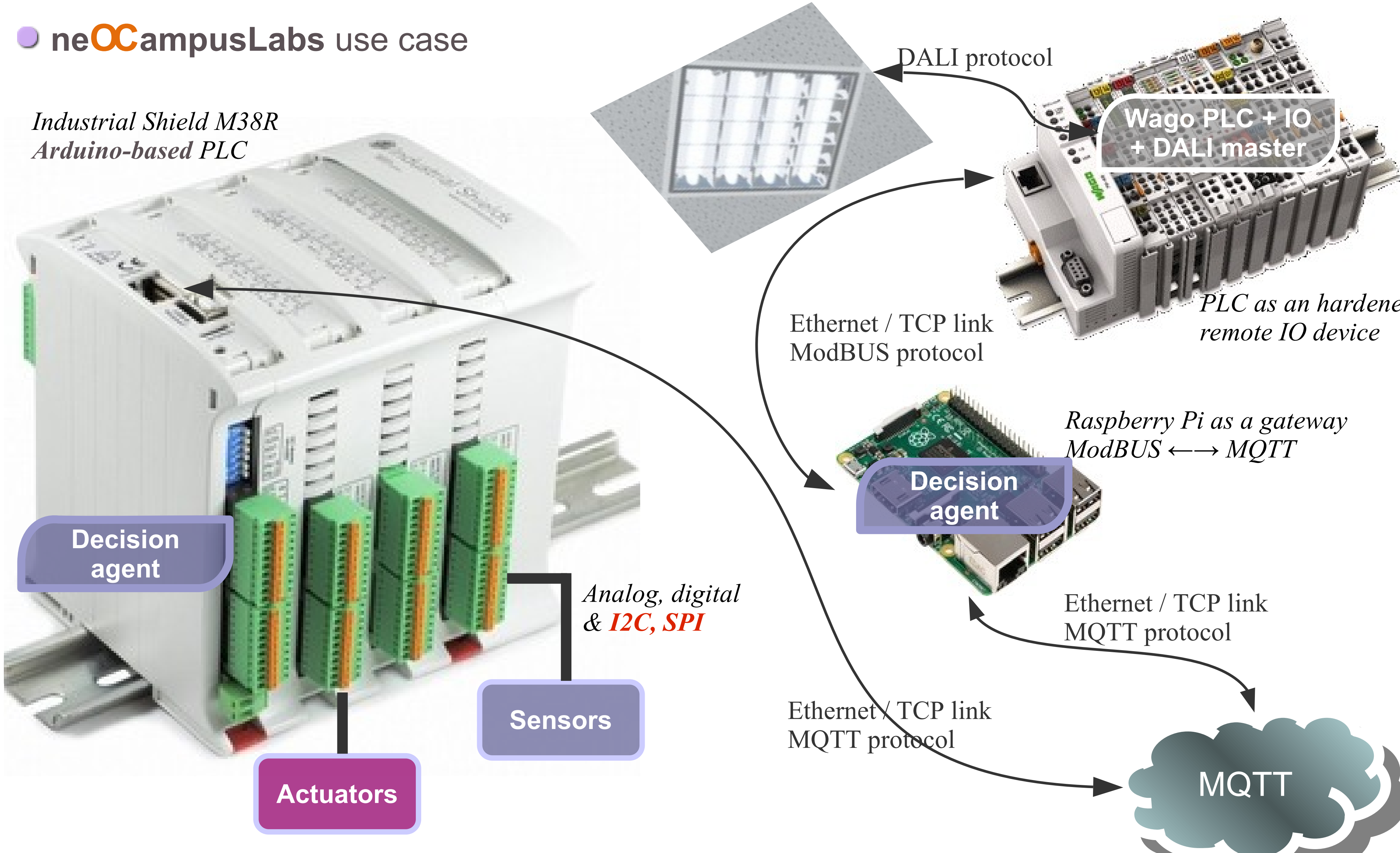
Overview | neOCampus

- **neOCampus** infrastructure



Overview | neOCampus

- **neOCampusLabs** use case



… a shift from Programmable Logic Controller (PLC) to devices from the Internet Of Things (IoT) !

Plan

Part I – 1 Principles

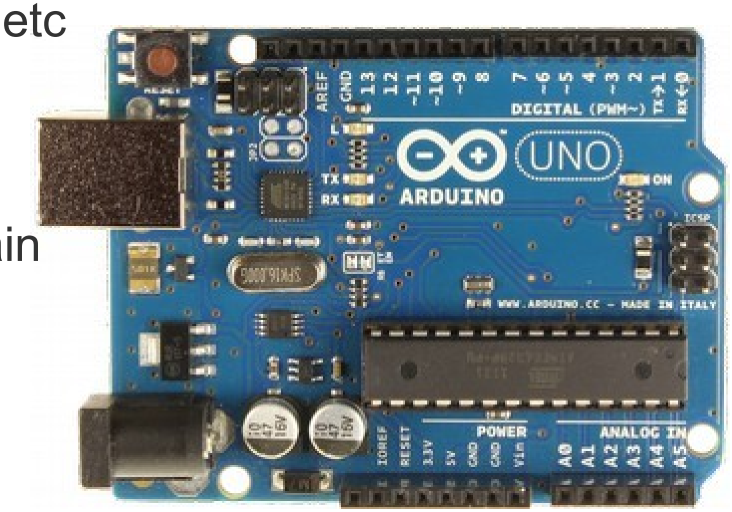
- Embedded Systems | Bare & OS-powered boards,
- Chip-level communications | I2C, SPI, Dallas 1-wire,
- Introduction to Raspberry Pi,
- Introduction to Python,
- Python @ Raspberry Pi | application to the I2C bus.

Embedded systems

- Bare boards

Arduino → Atmel* & Nordic chips + core libraries + IDE,
Nucléo → STM32 chips from STMicroelectronics,
PIC → famous µcontrolers from Microchip,
ARM boards from Atmel, Embedded Artist etc etc
…..........

** as of 2016, Microchip bought Atmel*



http://www.arduino.cc

Very (very) cheap (may requires only the main chip + some passives),
Simple and efficient libraries to use,
Full control over the execution runtime,
Low cost development boards.

Usually requires JTAG or some special link to update code,
Difficult to add, for example, a SSH server,
Mainly binary code (e.g. Python interpreter is very unlikely).

Embedded systems

- Smart Boards (OS-powered embedded systems):

Raspberry Pi → ARMv6, ARMv7 (Pi 2,3), ARMv8 (Pi 4)
Beaglebone → ARMv7 (Cortex A8)
RockChip RK3399 → dual Cortex A72 + Quad A53

Both are ARMv8-A 64bits, Cortex-A53: 2ways out-of-order, Cortex-A72: 3ways out-of-order

Low-cost solution (starting from 2012 for RPi & Beagleboard),

Fully featured systems with virtually everything (web server, java and python interpreters, DHCP server, SSH server …)

Full range of powerful libraries and bindings like I2C/SMBus for python,

Easy to update either your code or the system wherever you are.

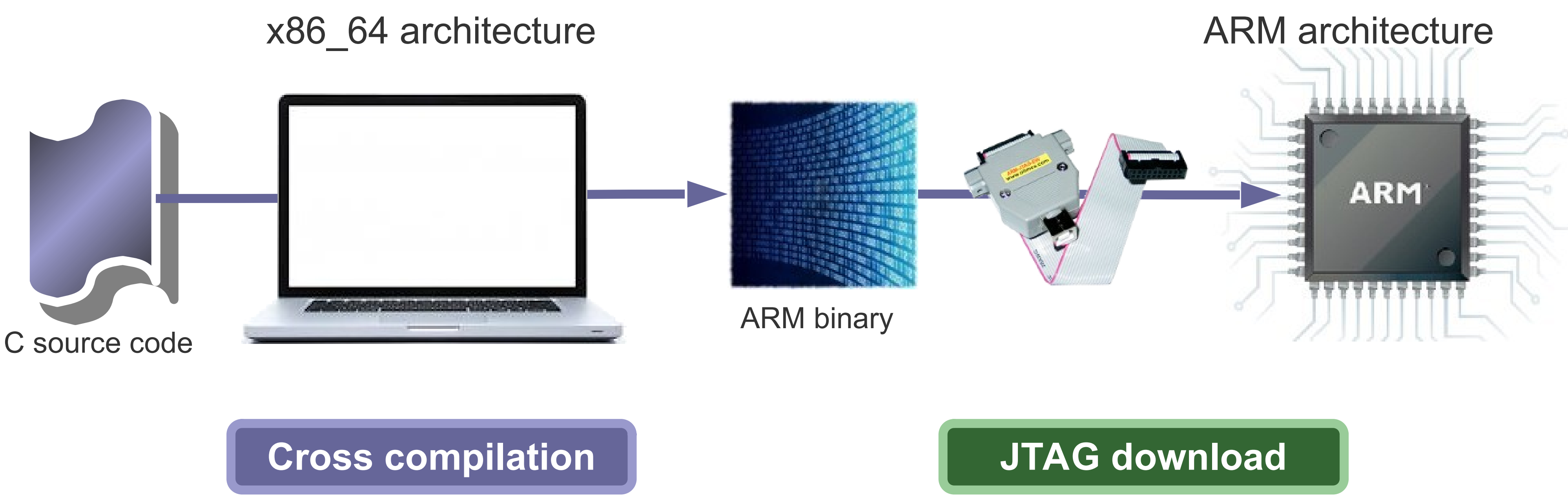
High complexity boards along with ASIC → impossible to build your own board unless you're a professional,

Except with real-time kernel, execution runtime will experience variability.

Cross Compilation

- Generate code for bare boards: cross compilation tool-chain

    Build binaries for an ARM architecture from a x86_64 computer.



Cross Compilation

- Example: generate binaries for ARM

Note: we do not use any of the default C libraries nor the low-level initialisation assembly code.

To download code through JTAG, generated binary may need to be converted to either DWARF, S3 or ELF format:

```
#> arm-eabi-objdump <options> <file>
```

Test.c

```c
void _start(int argc, char**argv) {

    // call to main
//  main(argc,argv);

    int i=10;
    int j=1;

    while (i-->=0) {
        j*=2;
    }

//  return j;

    while (1);
}
```

```
08:50:52 **** Build of configuration Debug for project simpleTest ****
make all
Building file: ../test.c
Invoking: Cross GCC Compiler
arm-eabi-gcc -O0 -g3 -Wall -c -fmessage-length=0 -MMD -MP -MF"test.d" -MT"test.d" -o "test.o" "../test.c"
Finished building: ../test.c

Building target: simpleTest
Invoking: Cross GCC Linker
arm-eabi-gcc -nostartfiles -nodefaultlibs -nostdlib -o "simpleTest"  ./test.o
Finished building target: simpleTest


08:50:54 Build Finished (took 1s.374ms)
```

Smart boards

- Executing code on a smart board

Launch Python code through a SSH connexion to a distant board.

```
thiebolt@metis[~] ssh root@raspicam1
Linux raspberrypi 4.14.67-v7+ #1139 SMP Wed Aug 29 15:17:05 BST 2018 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Thu Jul 19 16:43:52 2018 from 192.168.0.254
root@raspicam1[~] l
total 76K
8.0K -rwxr-xr-x  1 root root 4.5K Jun 21 15:57 Adafruit_I2C.py
4.0K -rw-r--r--  1 root root 2.2K Sep 29 23:24 enable_second_I2C.py
4.0K -rwxr-xr-x  1 root root 2.1K Oct  3 14:39 recs_temp_sensors.py
 20K -rwxr-xr-x  1 root root  18K Oct  9 00:38 temp_zabbix_RECS.py
root@raspicam1[~] ./recs_temp_sensors.py
I2C: Wrote 0x20 to register 0x01
I2C: Device 0x4D returned the following from reg 0x00
[21, 128]
Current temperature is 21.50°c ... conversion took 3ms
root@raspicam1[~]
```

```
root@raspicam1[~] cat /proc/cpuinfo
processor        : 0
...........
processor        : 3
model name       : ARMv7 Processor rev 4 (v7l)
BogoMIPS         : 38.40
Features         : half thumb fastmult vfp edsp
neon vfpv3 tls vfpv4 idiva idivt vfpd32 lpae
evtstrm crc32
CPU implementer : 0x41
CPU architecture: 7
CPU variant      : 0x0
CPU part         : 0xd03
CPU revision     : 4

Hardware         : BCM2835
Revision         : a02082
Serial           : 00000000fdbc97ba
```

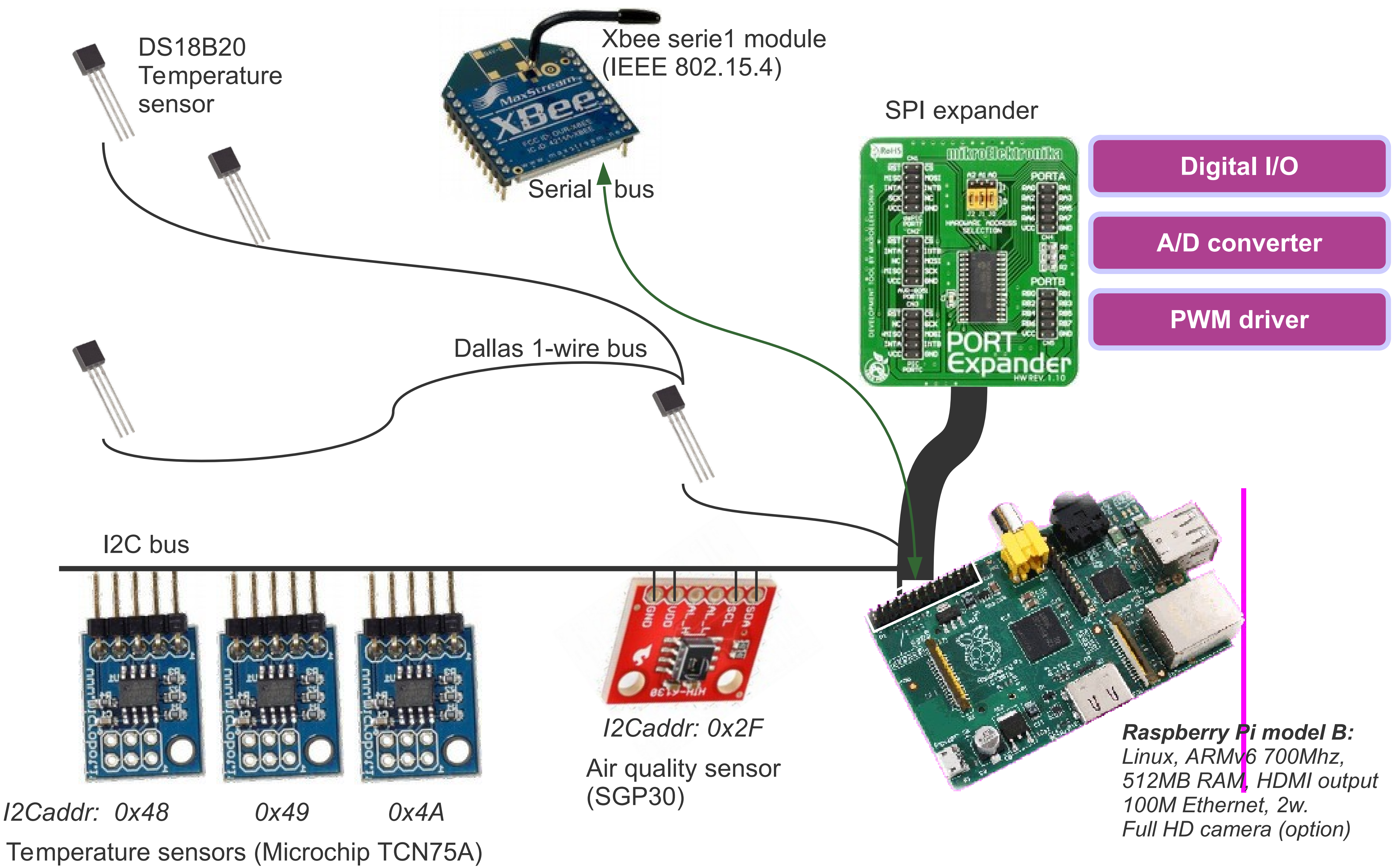
```
root@raspicam1[~] cat /proc/meminfo
MemTotal:         880424 kB
MemFree:          334224 kB
MemAvailable:     687920 kB
Buffers:           24012 kB
Cached:           380692 kB
...........
```

Plan

Part I – 1 Principles

- Embedded Systems | Bare & OS-powered boards,
- Chip-level communications | I2C, SPI, Dallas 1-wire,
- Introduction to Raspberry Pi,
- Introduction to Python,
- Python @ Raspberry Pi | application to the I2C bus.

Chip-level buses

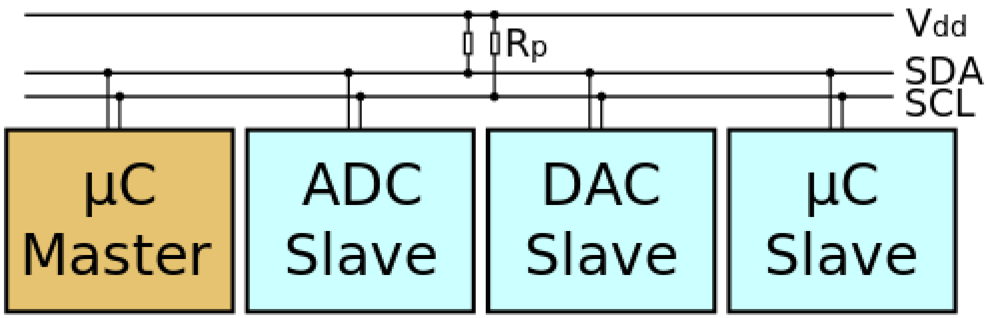


I2C bus

- Inter-Integrated Circuit, a chip to chip communication bus ...
  - Designed by Phillips in the 80's | first version **100KHz** in 1982,
  - Master / Slaves relationship,
  - Up to 128 slaves,
  - Serial bus, 2 wires: SDA & SCL with pull-up resistors,
  - Half duplex,
  - VCC ranges from 2.7v to 5.0v (RPi is only 3.3v!),
  - Additional speeds: 400 kbit/s Fast mode (1992), 1 Mbit/s Fast mode plus or Fm+, and 3.4 Mbit/s High Speed mode,
  - Partially user-configurable devices address,
  - 400 pf max. capacitance; e.g cat.5e cable 52pf/m → up to 7 meters.

http://www.framboise314.fr/i2c-plus-de-30-ans-et-toujours-dactualite/

I2C bus



Typical I2C devices interconnect.

- SMBus: a lightweight I2C compatible protocol

• SMBus defined by intel in 1995 is derived from I2C.

• SMBus is a single-ended simple two-wire bus for the purpose of lightweight communication.

• SMBus' clock frequency ranges from 10 kHz to 100 kHz. Voltage levels and timings are I2C compatible hence I2C and SMBus devices are often successfully mixed on the same bus.

• Most of the time, micro-controllers feature an SMBus with extended capabilities that looks like an I2Cbus. It is commonly named I2C/SMBus.

I2C bus

- I2C devices usually features 1 to 3 bits (sometimes 4 bits) for user-configurable device address.
- Others address bits are factory-programmed.

- Example: TCN75A, an I2C temperature sensor

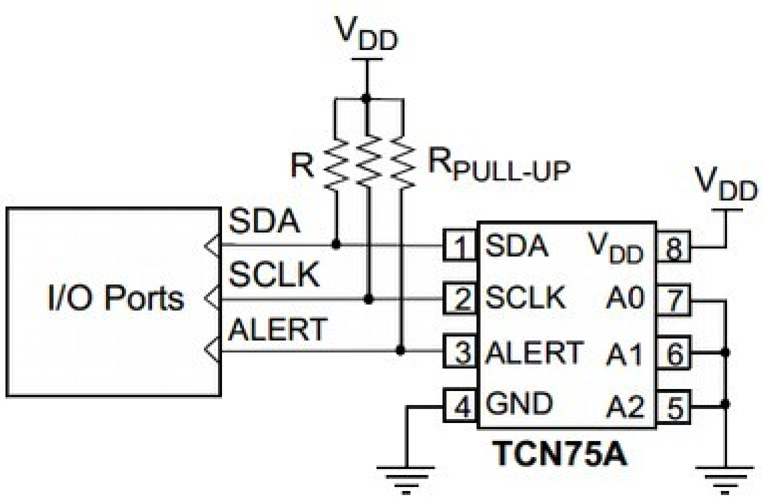


TABLE 3-2: SLAVE ADDRESS

Device	A6	A5	A4	A3	A2	A1	A0
TCN75A	1	0	0	1	X	X	X

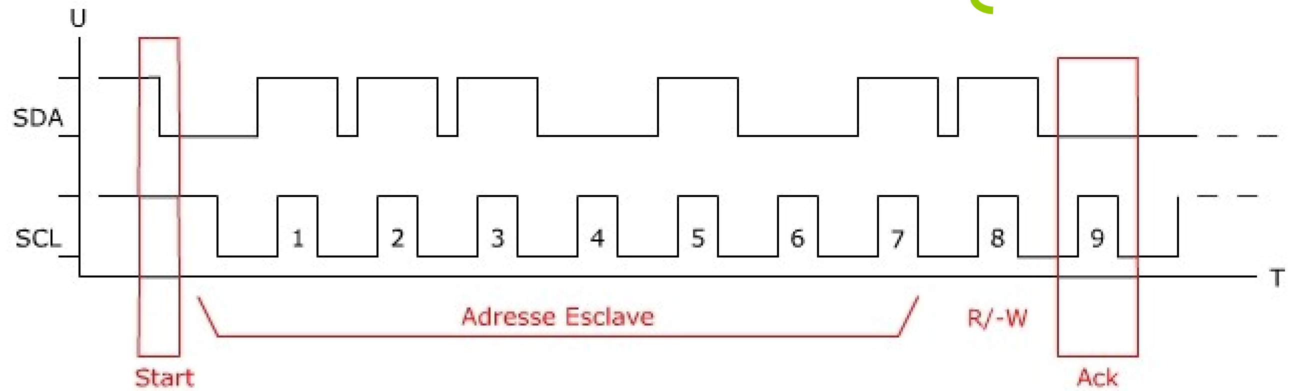
Note: User-selectable address is shown by X.

- Up to 8 devices,
- I2C Addresses range from 0x48 to 0x4F.

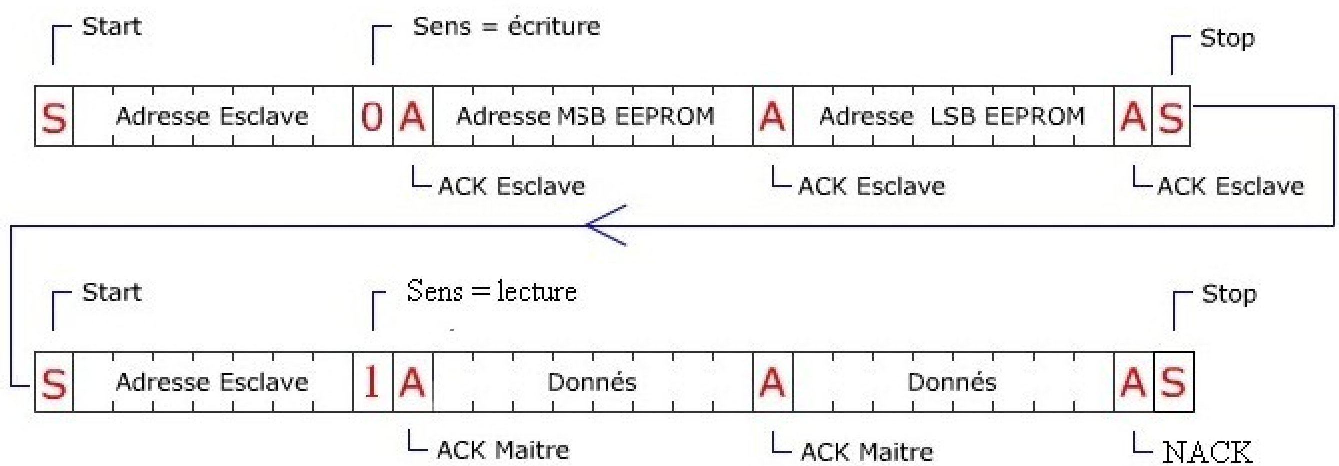
I2C bus

- Sample I2C transactions
  - I2C device addr = 7 bits + 1 RW bit

0 → write
1 → read



  - Protocol: Indirect access to registers within device, e.g read from an eeprom.



SPI bus

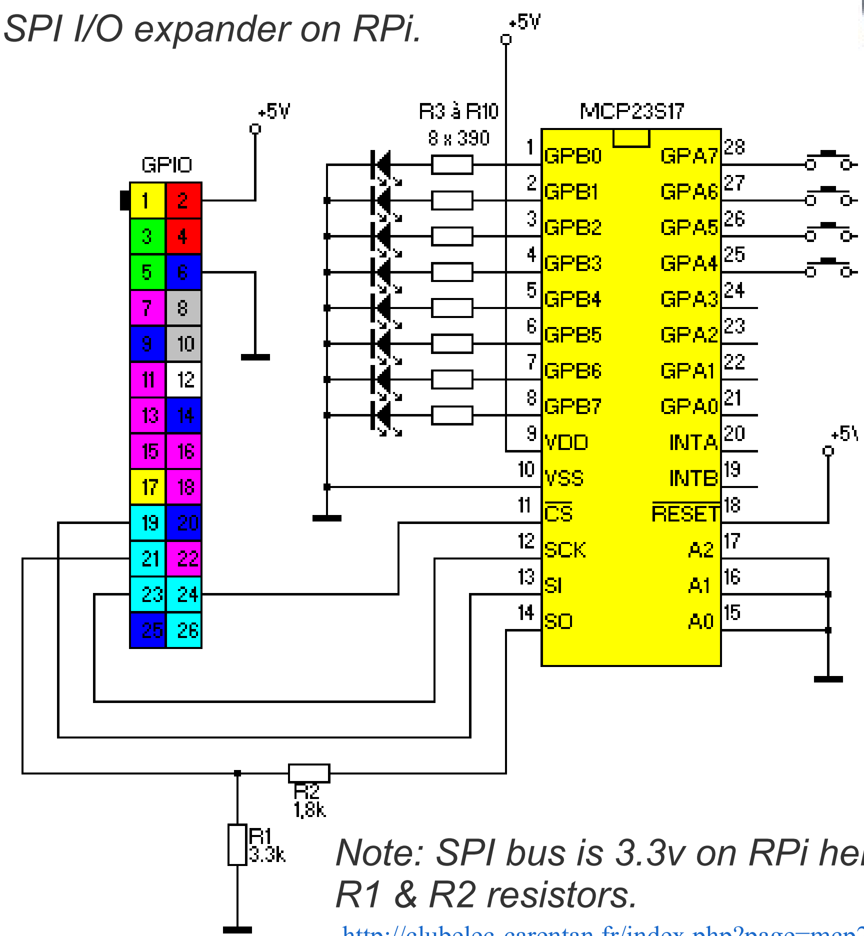
- Serial Peripheral Interface
  - Like I2C, designed by the beginning of the 80's but for higher throughput,
  - Master / Slaves relationship,
  - Serial bus with 3 wires: MISO, MOSI & SCLK,
  - Full duplex,
  - VCC ranges from 2.7v to 5.0v (RPi is only 3.3v!),
  - Intended to PCB (short wires),
  - On Raspberry Pi, SPI speed = APB* core clock (250 Mhz) / 2 to 32768,
  - Chip Select pins (CS) are generated by the Master to select one device at time,
  - On Raspberry Pi, 2 CS signals available (`ss0`, `ss1`).

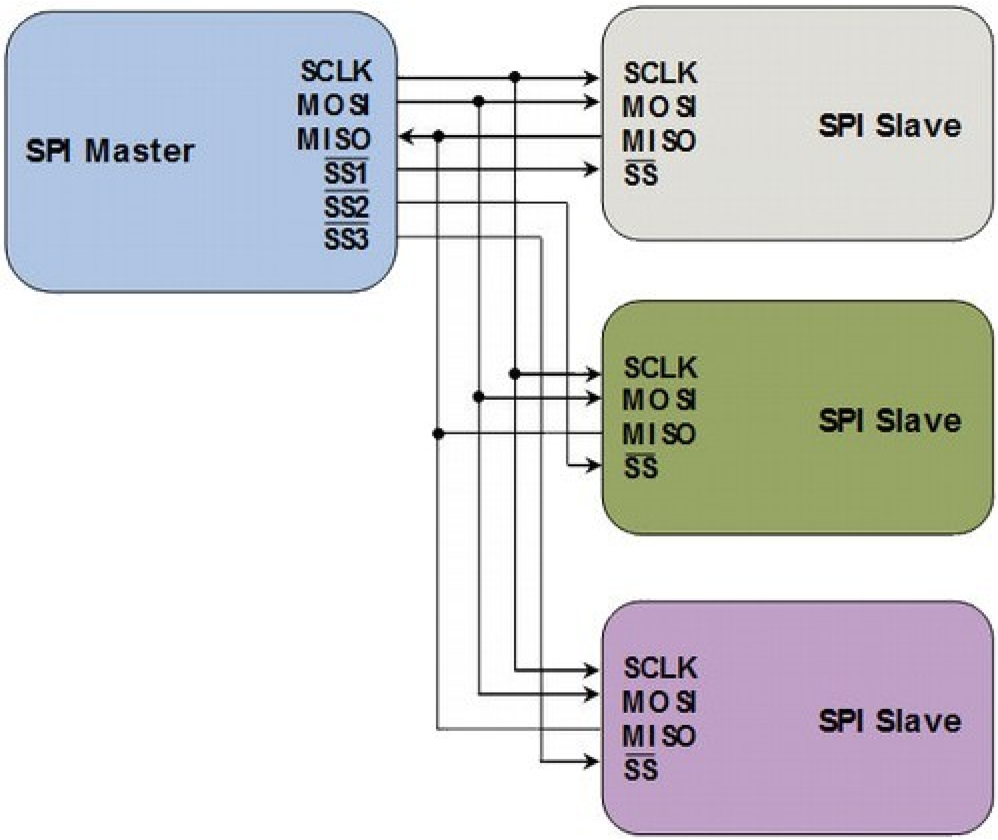
**ARM Peripheral Bus*

http://www.byteparadigm.com/applications/introduction-to-i2c-and-spi-protocols/

SPI bus

SPI I/O expander on RPi.





Typical SPI devices interconnect.

Note: SPI bus is 3.3v on RPi hence the R1 & R2 resistors.

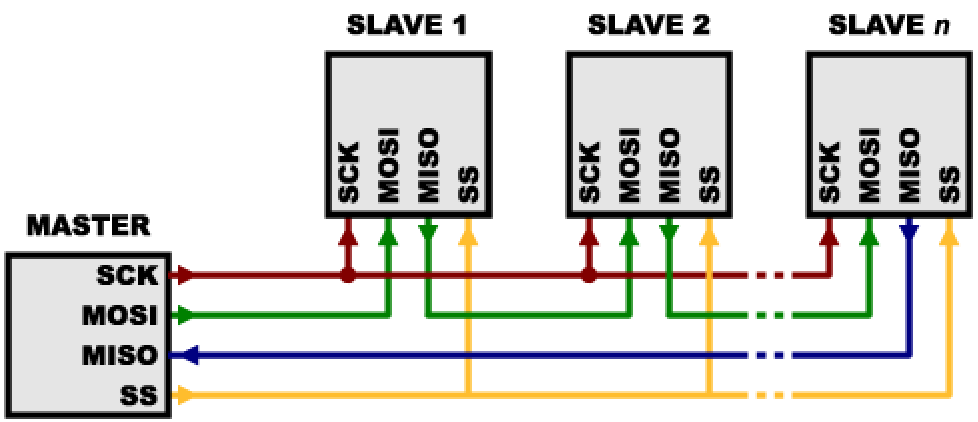
http://clubelec-carentan.fr/index.php?page=mcp23s17

SPI bus

- Each SPI device requires a dedicated CS line,
- Things become tricky when having a large number of SPI devices …

  (MCP23S17 with addressing solve this issue)

SPI daisy-chain mode



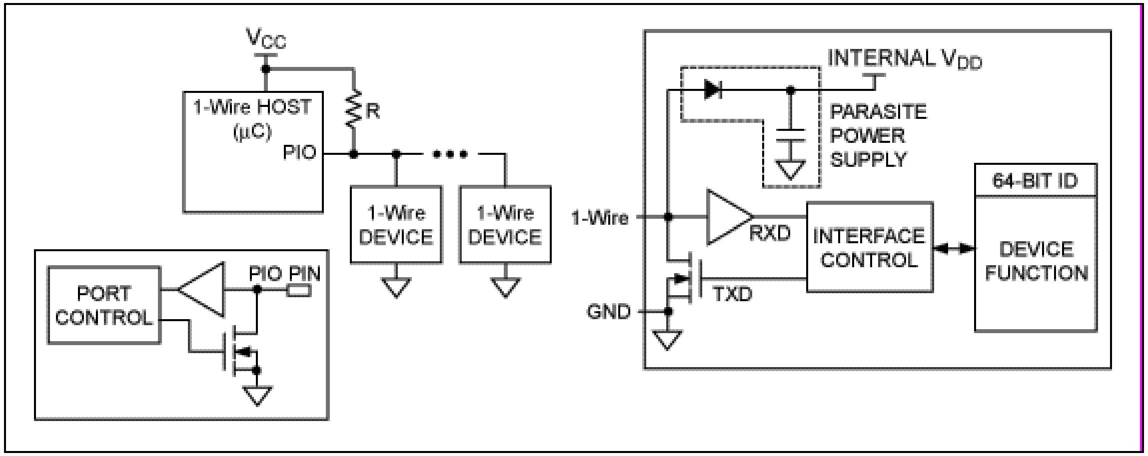
- Only one CS line for multiple slaves,
- Rely on slave's internal shift registers which propagates MOSI → MISO as long as CS remains active,
- On rising-edge of CS, each slave executes the command in its input buffer. This way, all slaves may execute a different command.

Infineon: SPI interface used in daisy-chain
https://learn.sparkfun.com/tutorials/serial-peripheral-interface-spi/slave-select-ss

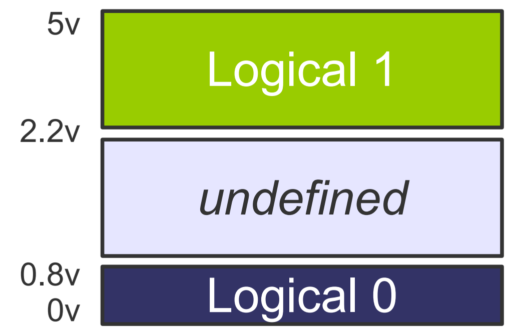
- Dallas 1-wire bus
  - Designed by Dallas Semiconductor (now Maxim),
  - Lower data rates than I2C but longer range,
  - Typically used to communicate with low cost devices like temperature sensors,
  - 1-wire devices include an 800 pF capacitor to store charge, and power the device during periods when the data line is active (parasitic mode),
  - Pull-up resistor on data line to power devices. Devices and Master exhibit an open drain,
  - VCC ranges from 2.7v to 5.25v,
  - 15.4 kbps (standard) to 125 kbps (overdrive),
  - Slave device has a unique, unalterable, factory-programmed, 64-bit ID,
  - Use twisted cable,
  - Up to 200m with regular pull-up resistor and up to 500m with active termination.

http://www.maximintegrated.com/app-notes/index.mvp/id/148
http://www.maximintegrated.com/app-notes/index.mvp/id/1796

Dallas 1-wire bus

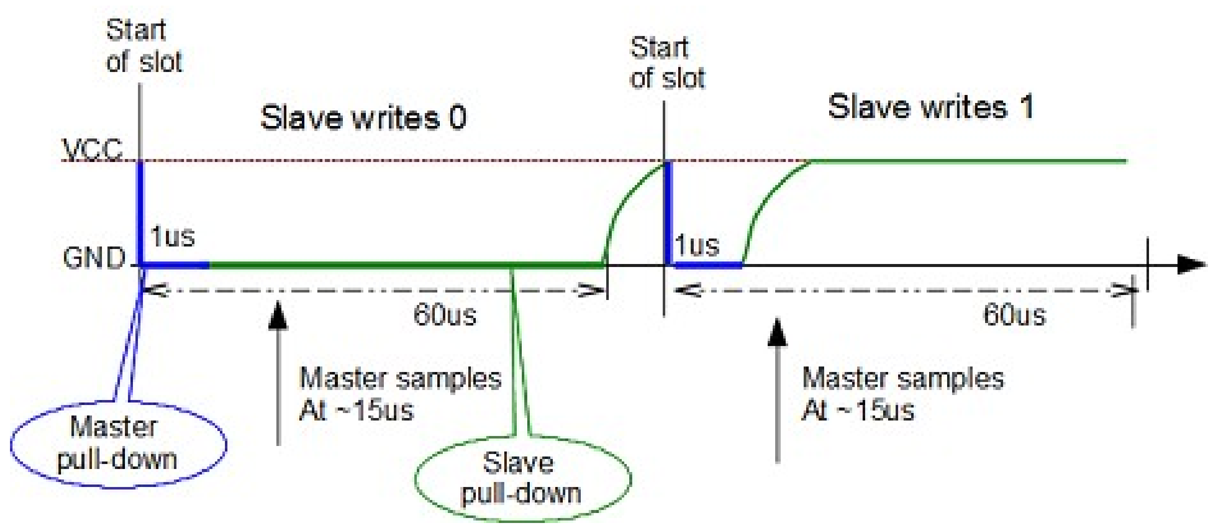


1-wire typical interconnect & slave internals.

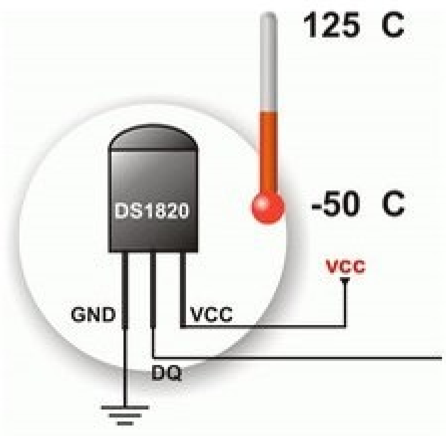


1-wire logical levels.

- Concept of time slot: 60µs (standard) & 8µs (overdrive),
- 0 and 1 are encoded within timeslot.



Dallas 1-wire bus



1-wire temperature sensor Dallas DS1820B.

iButton is 1-wire powered (eeprom).

- 1-Wire on Raspberry Pi

  There's no 1-Wire master device on RPi. Instead one GPIO pin + pull-up resistor is used to drive the line. The **w1-gpio** kernel module implement the 1-Wire protocol (Bit banging). http://blog.gegg.us/2013/03/4-different-methods-of-1-wire-access-on-raspberry-pi/

Chip-level buses: synthesis

- Chip-level buses synthesis

Name	Wires, Duplex	CLK / Speed	Length	Notes
I2C/SMBus	Two wires, Half duplex	10 → 100 KHz 400 KHz 1 MHz	Up to 7 meters	One master, up to 128 slaves, pull-up resistors
SPI	Three wires, Full duplex	Chip dependant, 10 Mhz std.	PCB intended	One master
Dallas 1-Wire	One wire, Half duplex	15.4 kbps (std) 125 kbps	May reach 500-700m with ad-hoc termination	One master, fixed devices addr.

... jump one step further with http://homepages.laas.fr/bvandepo/files/comm/cours_comm_numbered.pdf

Plan

Part I – 1 Principles

- Embedded Systems | Bare & OS-powered boards,
- Chip-level communications | I2C, SPI, Dallas 1-wire,
- Introduction to Raspberry Pi,
- Introduction to Python,
- Python @ Raspberry Pi | application to the I2C bus.

Raspberry Pi

[aug. 16]
10 million RPi sold!

- Raspberry Pi … a 2012 started revolution … and now **RPi4**
  - Fully-featured embedded system with Linux*, Android and even Win10!,
  - Powerful quad-cores 64bits ARMv8 Cortex A72@1,5GHz
    (Pi3B+ Quad A57 ARMv8@1.4GHz, single core ARMv6@1GHz for Pi0W)
  - GPU | H264/265 encode/decode → dual 4K grade displays and camera,
  - Ethernet Gigabit with PoE option via addon board,
  - Up to 4GB ram shared with GPU,
  - Plenty of available I/O, expansion shields** (e.g PiFace),
  - Promoted by a non-profit foundation,
  - #5W, huge community and still 35€ priced!
- but ...
  - Not totally open-hardware nor open-source,
  - blob driven GPU (Binary Large OBject),
  - No PXE boot (**until august 16 :)**).

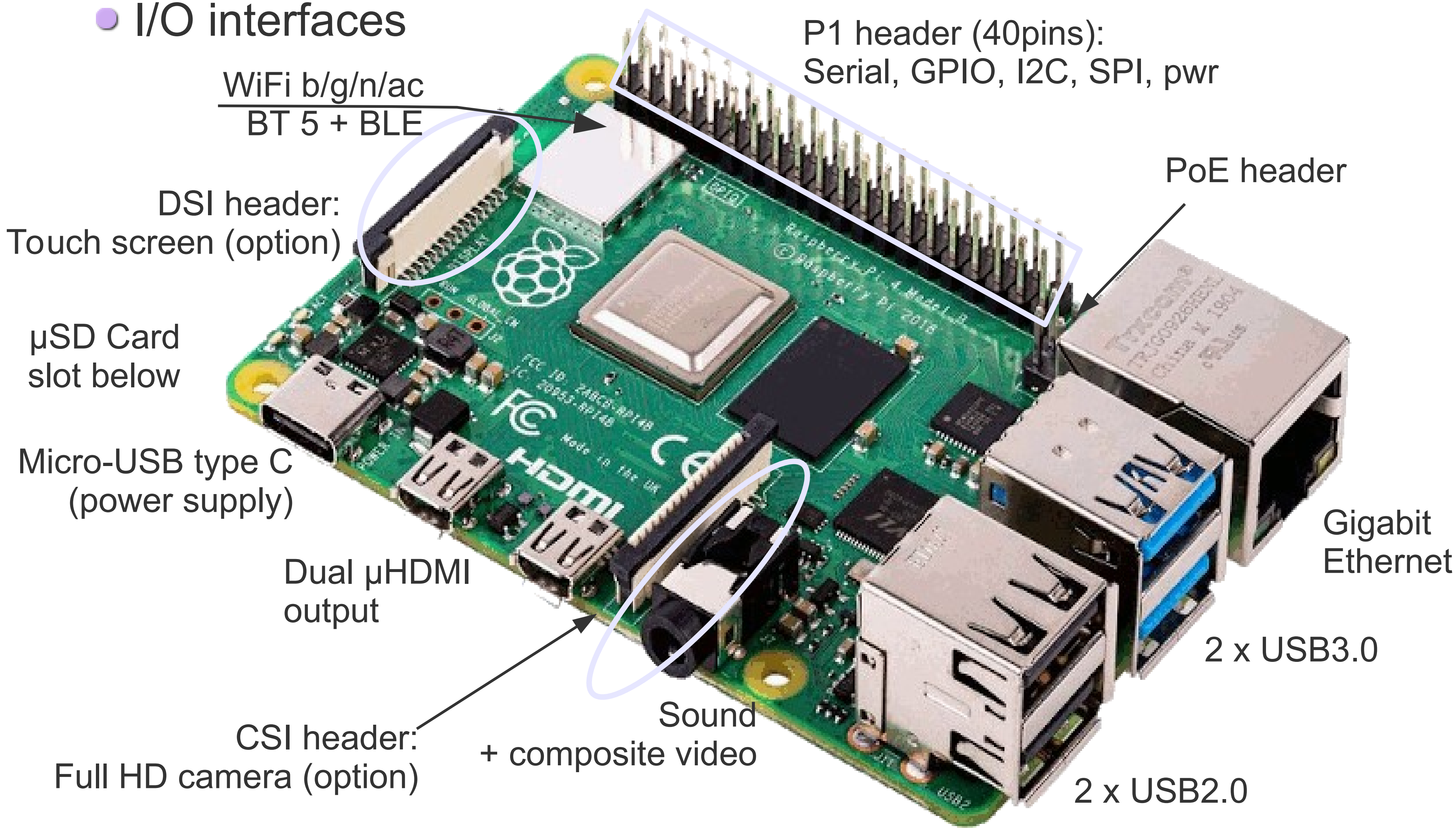
** there exists several Linux distributions for this board like Raspbian, Pidora, Arch Linux … and NOOBS which let you deciding what system to install (keyboard and HDMI display required).*

Unless otherwise specified, we'll use the Raspbian distribution (Linux Debian based on).
http://www.raspberrypi.org/downloads

*** arduino terminology*

Pi 4B (2019)

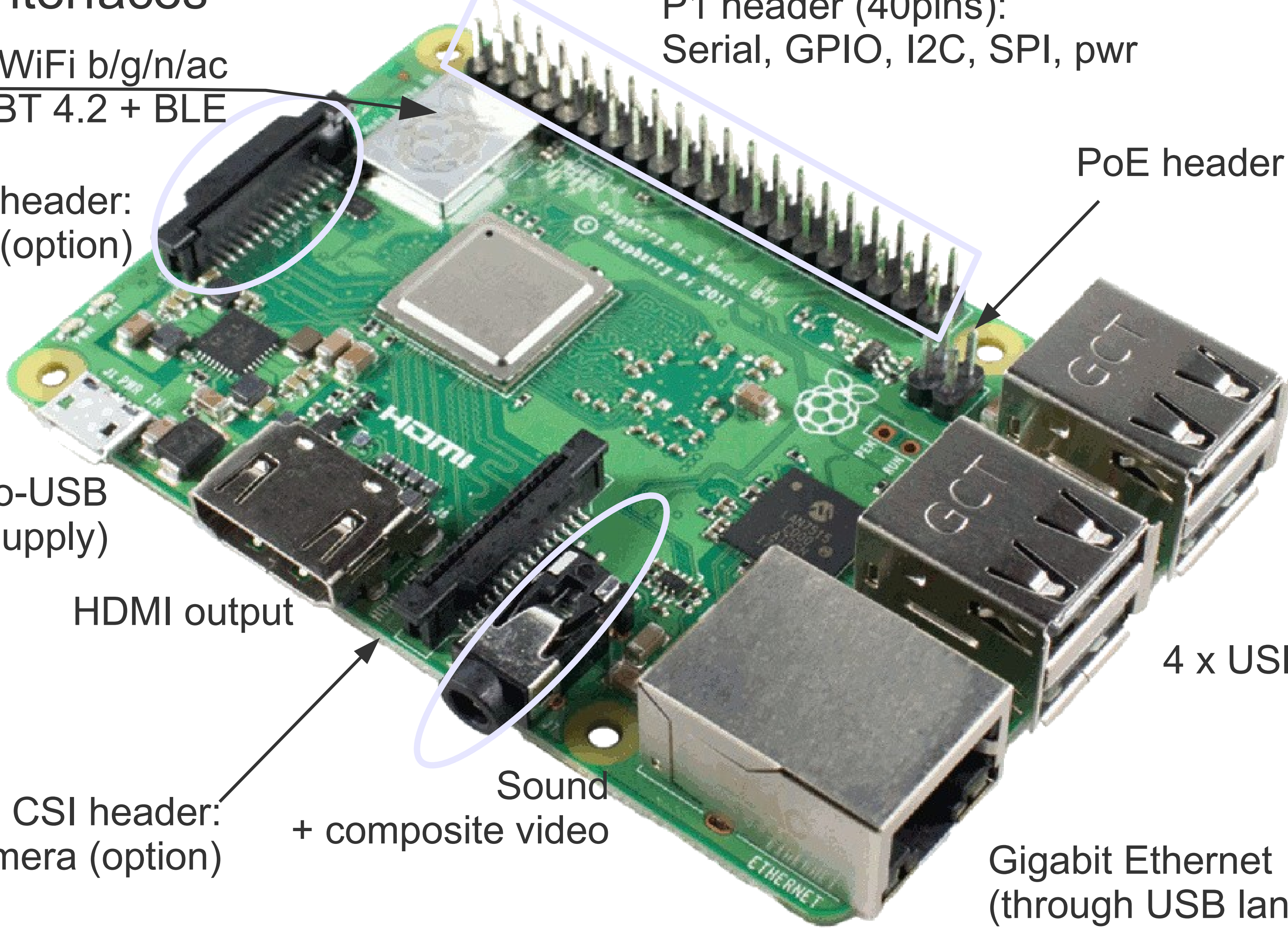
- I/O interfaces



[sep.19] RPi4b 4GB → #49€ HT

Raspberry Pi 3B+ (2017)

- I/O interfaces



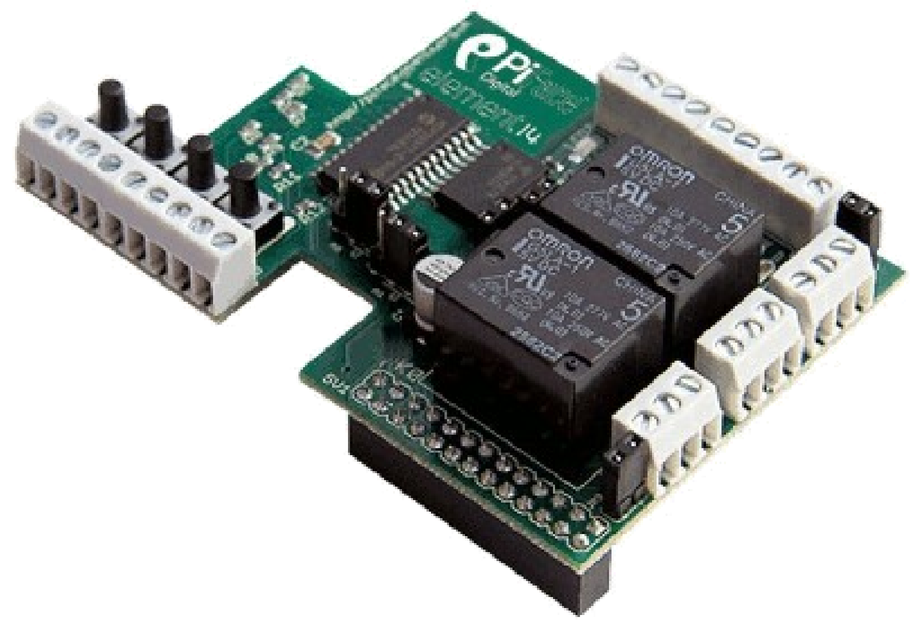
Dimensions: 85 x 56 x 17mm

Raspberry Pi accessories

- P1 header details

Pin#	NAME	NAME	Pin#
01	3.3v DC Power	DC Power 5v	02
03	GPIO02 (SDA1 , I²C)	DC Power 5v	04
05	GPIO03 (SCL1 , I²C)	Ground	06
07	GPIO04 (GPIO_GCLK)	(TXD0) GPIO14	08
09	Ground	(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)	(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)	Ground	14
15	GPIO22 (GPIO_GEN3)	(GPIO_GEN4) GPIO23	16
17	3.3v DC Power	(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)	Ground	20
21	GPIO09 (SPI_MISO)	(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)	(SPI_CE0_N) GPIO08	24
25	Ground	(SPI_CE1_N) GPIO07	26
27	ID_SD (I²C ID EEPROM)	(I²C ID EEPROM) ID_SC	28
29	GPIO05	Ground	30
31	GPIO06	GPIO12	32
33	GPIO13	Ground	34
35	GPIO19	GPIO16	36
37	GPIO26	GPIO20	38
39	Ground	GPIO21	40

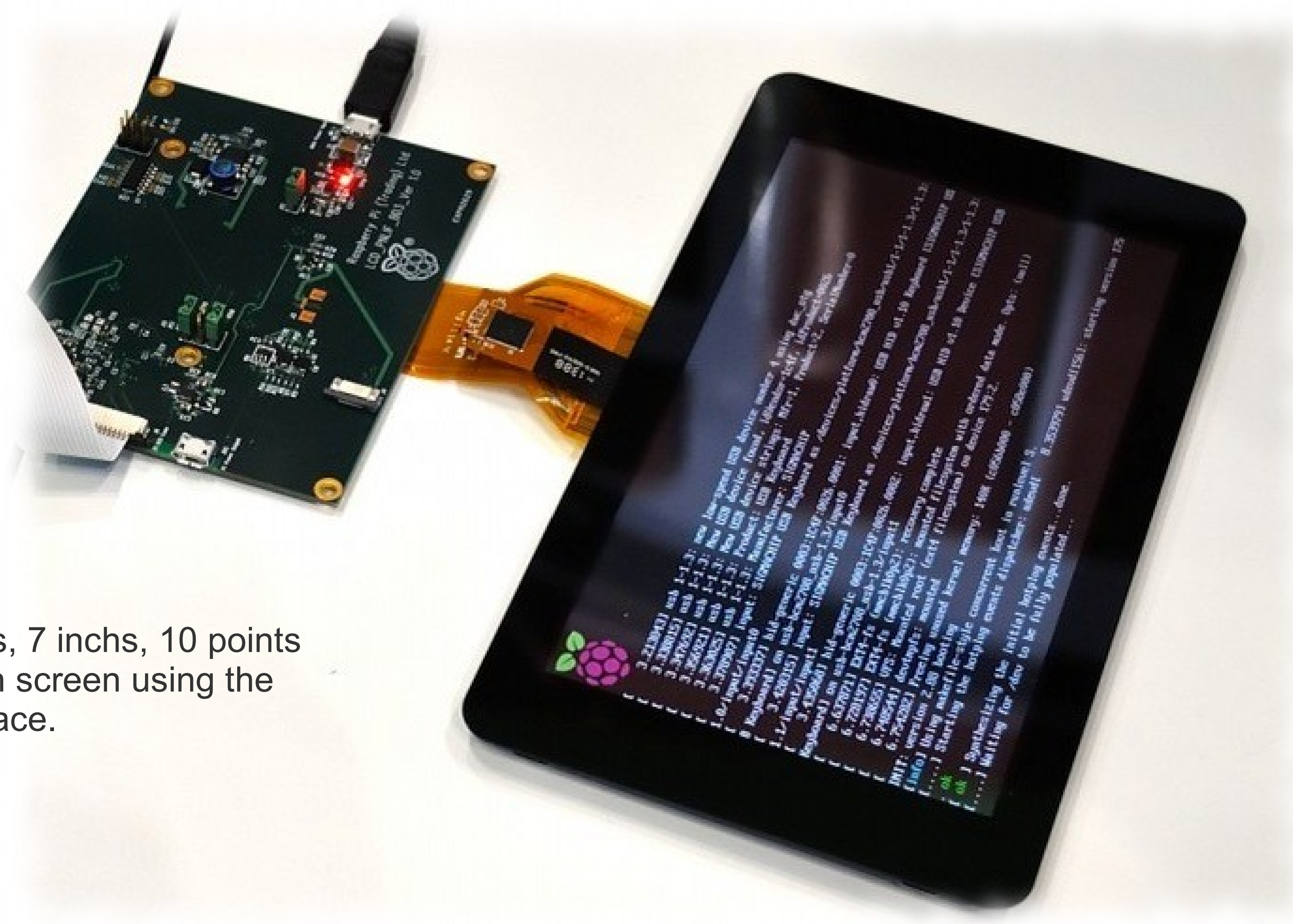
- PiFace expansion board



This board provides 4 digital input coupled with on-board switches along with 8 digital output coupled with on-board leds. Two of the eight outputs are also tied with 5v relay coils to drive 230v loads.

Raspberry Pi accessories

- DSI touchscreen



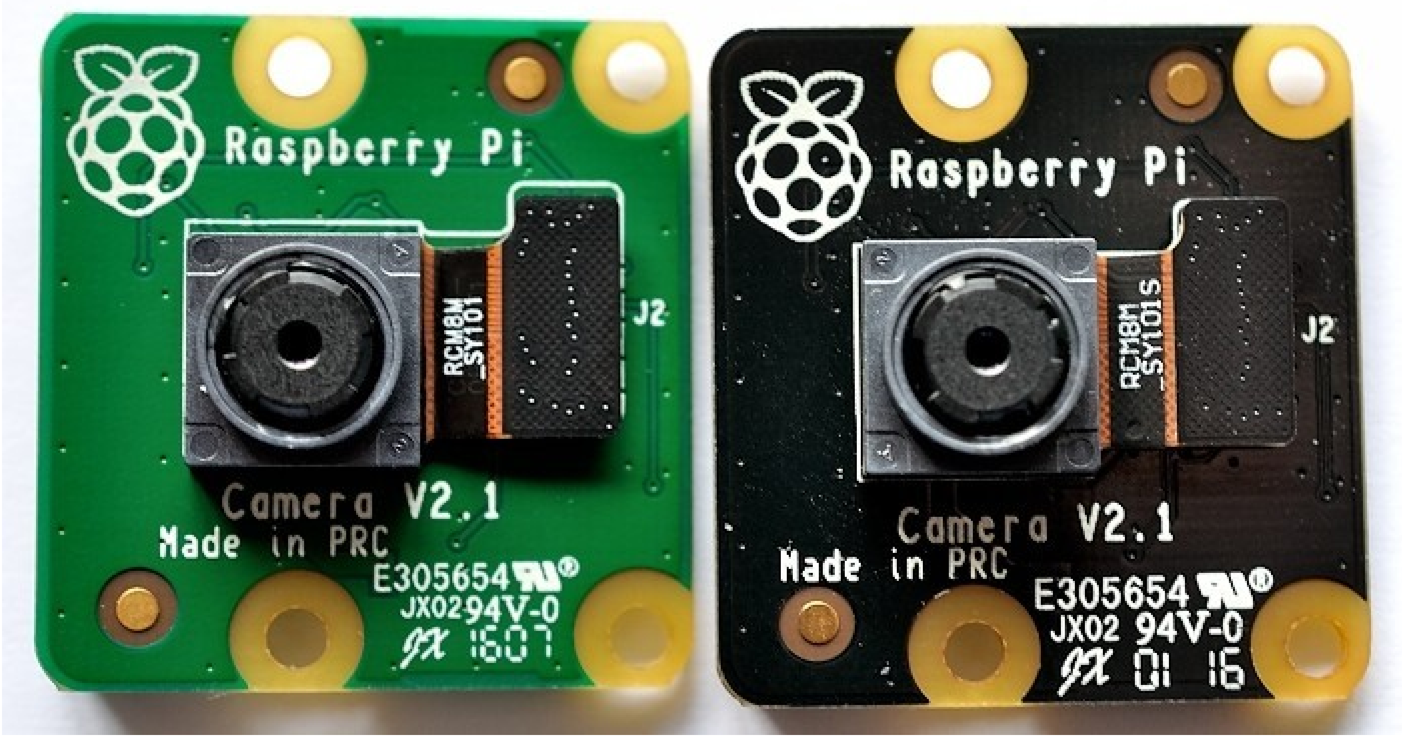
800 x 480 pixels, 7 inchs, 10 points capacitive touch screen using the RPi's DSI interface.

Raspberry Pi accessories

- Full-HD camera v2 1920x1080p 30fps,
- Still pictures 3280 x 2464 (8MP)

Standard version

Pi NoIR version

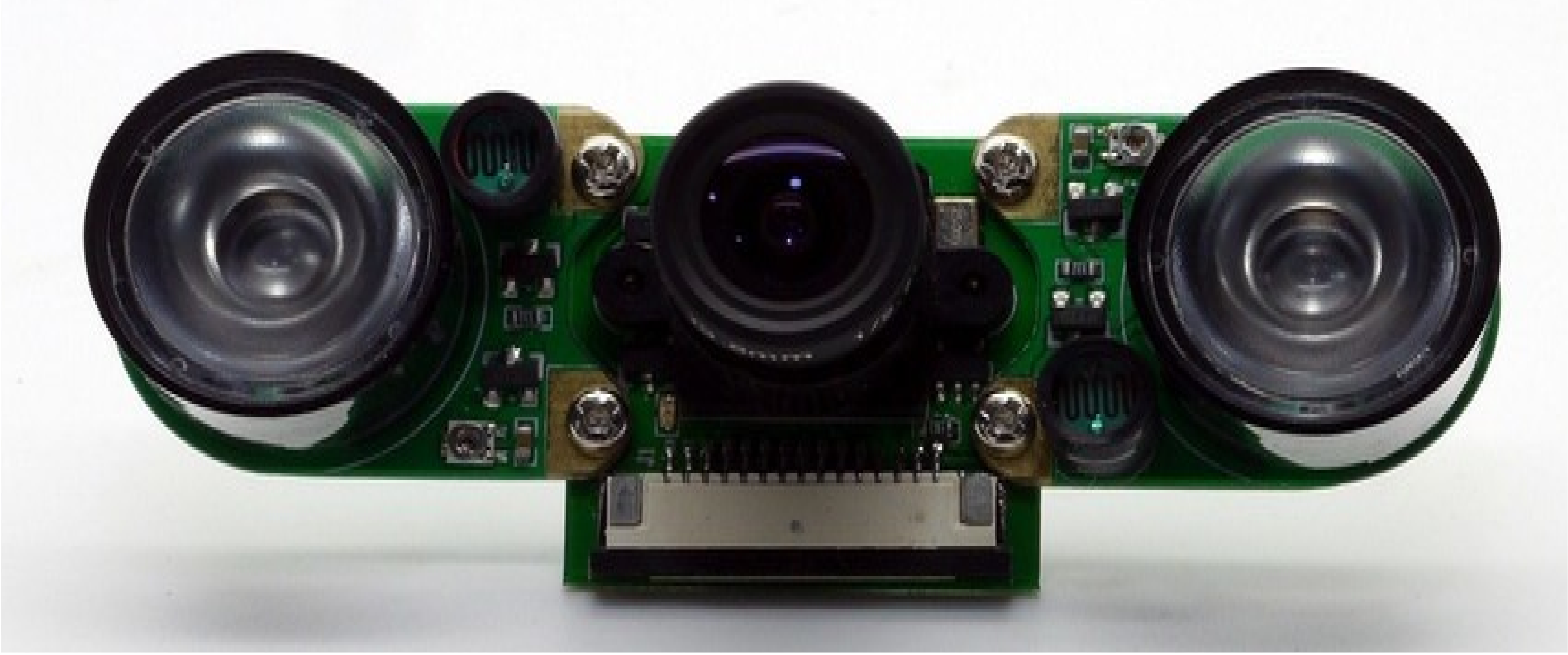


No IR filter in this version: enables night vision with the use of power IR leds.

Possible slow motion 640 x 480p 90fps. Dimensions: 24 x 25mm

Raspberry Pi accessories

- PiNoIR with wide lens + 2 x IR leds featuring light sensor



Raspberry Pi

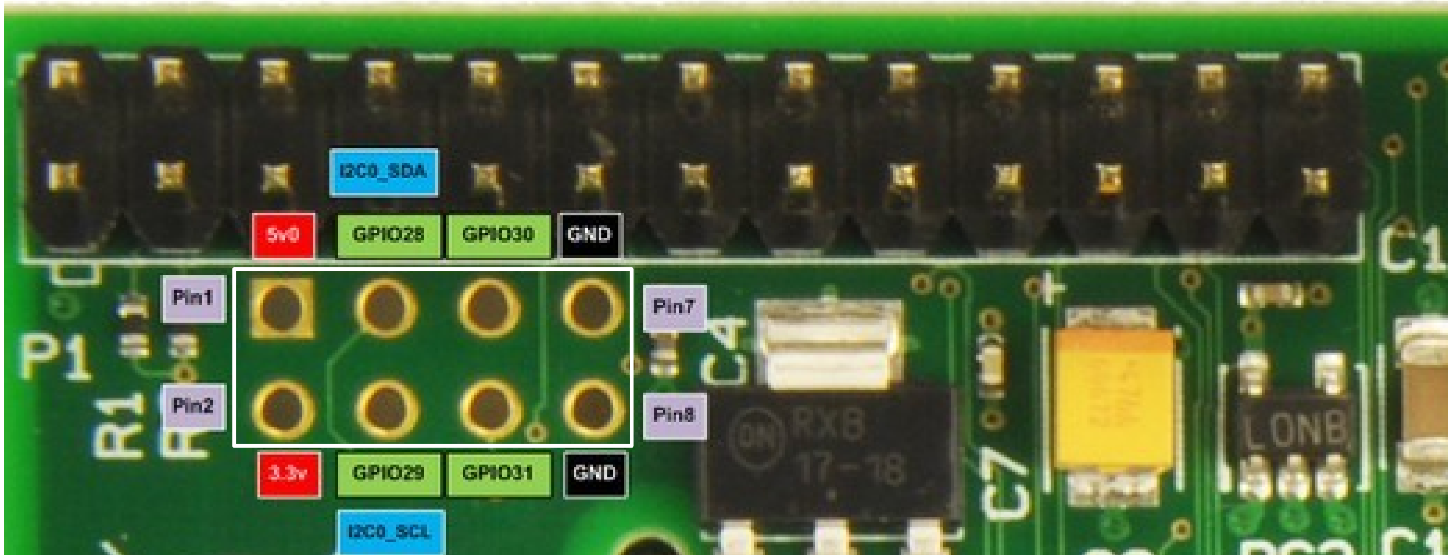
- Board configuration | the `raspi-config` command

  - Enable / disable camera support,
  - Set hostname,
  - Password change,
  - Boot options,
  - Extend filesystem up to the whole SDcard,
  - Enable / disable SSH server [default **OFF**],
  - Overclocking,
  - CPU | GPU memory split,
  - ...

Note: the `raspi-config` command comes from the Linux Raspbian distribution.

- Expanding Raspberry Pi GPIO (version 1) ...

  … and uncover the second I2C bus on the unpopulated P5 header!



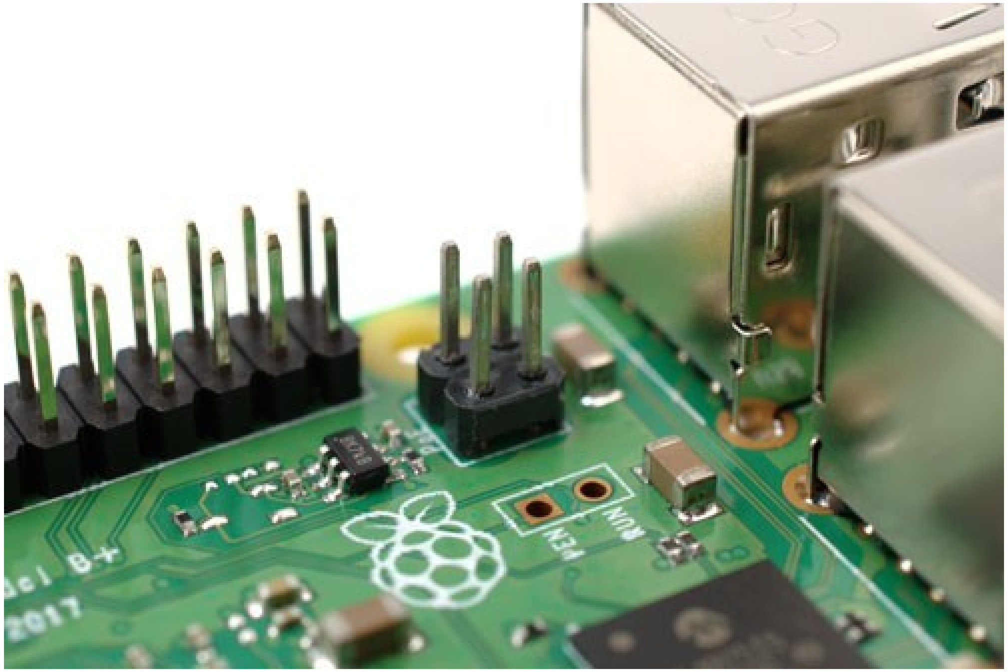
Raspberry Pi model B rev. 2.0 | unpopulated P5 header.

http://developmentboards.blogspot.fr/2013/04/expand-number-of-gpio-pins-on-raspberry.html

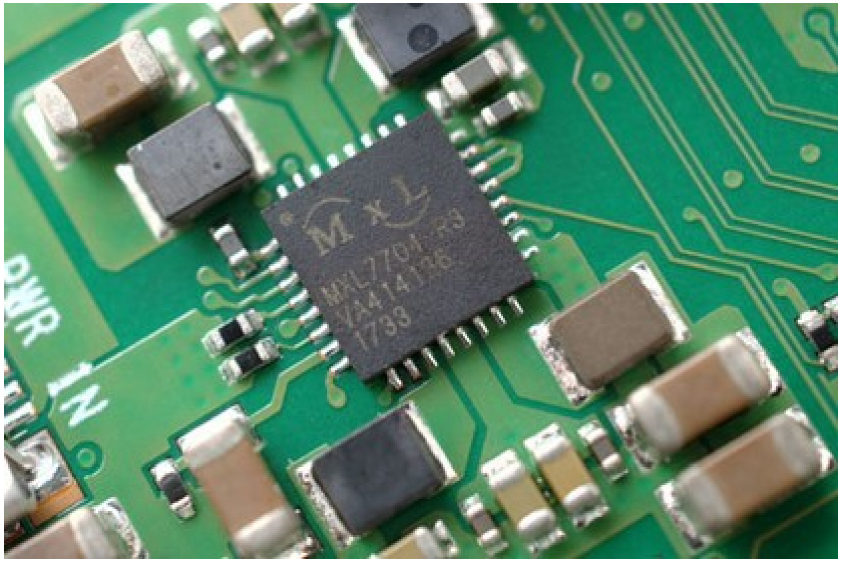
[Advanced] Raspberry Pi

- PEN and RUN pins @ RPi3B+ and RPi4 ...

  … towards a real hardware watchdog ?? (i.e not a system one)



RPi3B+, PoweENable and RUN pins …



RPi3B+ and RPi4 use PMIC MXL7704 …

Plan

Part I – 1 Principles

- Embedded Systems | Bare & OS-powered boards,
- Chip-level communications | I2C, SPI, Dallas 1-wire,
- Introduction to Raspberry Pi,
- Introduction to Python,
- Python @ Raspberry Pi | application to the I2C bus.

Python

- Introduction to the Python language
  - Open-source general-purpose language,
  - Object-oriented, procedural, functional,
  - Interpreted language (i.e not for bare boards),
  - Widely available for almost all platforms (Mac, Windows, Linux etc...),
  - Available python interactive interpreter,
  - Some changes between Python 2.x and 3.x versions (3.x ONLY nowadays).

Type in a shell ...

```
ssh -p 2220 g2ebi@m2siame.univ-tlse3.fr
Passwd: <secret>
g2ebi@frontal[~] python3
Python 3.7.4 (default, Jul  9 2019, 16:48:28)
[GCC 8.3.1 20190223 (Red Hat 8.3.1-2)] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

Python scripts

To execute python code directly from files:

1. create a script file named `<xxx>.py`,
2. add it execution capability bit.

```
touch script.py
chmod a+x ./script.py
```

A Python script ought to contain these first two lines

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
#
<code>
```

Alternatively, first line may be `#!/usr/bin/python`

… let's switch to Python tutorial

http://m2siame.univ-tlse3.fr/teaching/francois/G2EBI_python_tutorial.pdf

Python exercises

- Some simple problems to be solved with Python ...

Bubble sort

```
raw_data = [ 52, 17, 23, 5, 19, 4 ]
output_list = [ 4, 5, 17, 19, 23, 52 ]
```

Sum of the multiples of 3 or 5

```
If we list all the natural numbers below 10 that are multiples of 3 or 5, we get 3, 5, 6 and 9.
The sum of these multiples is 23.

Find the sum of all the multiples of 3 or 5 below 1000.
```

Fibonacci number

```
Each element is the sum of the previous two. List starts with 0 and 1:
Fibonacci = [ 0, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55 89, 144, ... ]

Display Fibonacci number up to 2000.
```

Mathematical problems to solve with a computer http://projecteuler.com

- Timer in Python | Introduction to multi-threading

… or the art of running simultaneous instances of a same portion of code

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-
#

import threading

def do_every (interval, worker_func, iterations = 0):
    if iterations != 1:
        threading.Timer (
                        interval,
                        do_every, [interval, worker_func, 0 if iterations == 0 else iterations-1]
                        ).start ();

    worker_func ();

def print_hw ():
    print "hello world";

def print_so ():
    print "stackoverflow"


# call print_so every second, 5 times total
do_every (1, print_so, 5);

# call print_hw two times per second, forever
do_every (0.5, print_hw);
```

Plan

Part I – 1 Principles

- Embedded Systems | Bare & OS-powered boards,
- Chip-level communications | I2C, SPI, Dallas 1-wire,
- Introduction to Raspberry Pi,
- Introduction to Python,
- Python @ Raspberry Pi | application to the I2C bus.

- an I2C bus scanner

… we'll make use of the write_quick I2C method

```
#!/usr/bin/env python3
# -*- coding: utf-8 -*-

import smbus

# Function to scan an i2c bus
def i2cScan(busnum, debug=False):
    # Range of possible I2C addr
    _I2C_ADDR_RANGE = range( 0x3, 0x78)

    # instantiate bus
    try:
        bus = smbus.SMBus(busnum)
    except IOError as err:
        if (debug): print("[%s] i2c busnum '%d' does not seem to exist!" % (__name__,busnum))
        return
    devicesList = []
    # parse addr list
    for addr in _I2C_ADDR_RANGE:
        try:
            ret=bus.write_quick(addr)
        except IOError as err:
            continue
        if (debug): print("[0x%X]" % addr, end="")
        devicesList.append(addr)
    return devicesList
```

```
def main():
    # i2c bus scan
    i2cbus=1;
    print("I2C bus scan (default bus): ")
    _i2cList = i2cScan(i2cbus, debug=True)
    if (_i2cList != None):
        print("\tFound %d I2C devices\n" % len(_i2cList))
    else:
        print("\n\tI2C bus %d does not exist\n" %i2cbus)

# Execution or import
if __name__ == "__main__":

    # Start executing
    main()
```

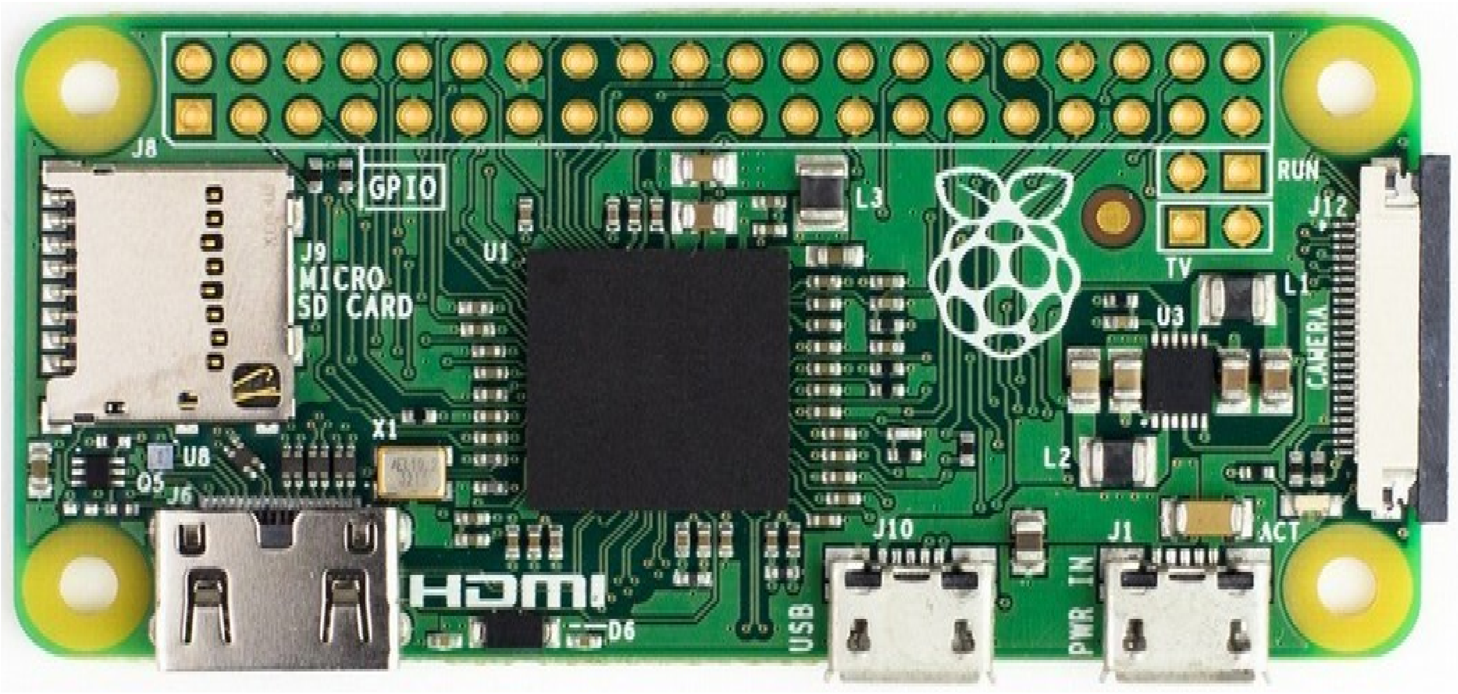
Plan

Part I – 2 SBC, communication chips ...

- Single Board Computers,
- Communication modules,
- Industrial modules,

Raspberry Pi Zero

- [Mar.17] Pi 0 W with WiFi and BT4.2 … 11€ priced
- [Feb.16] Pi 0 v1.3 with camera support … still $5!
- [Nov.15] Pi 0: a new revolution … $5 priced!



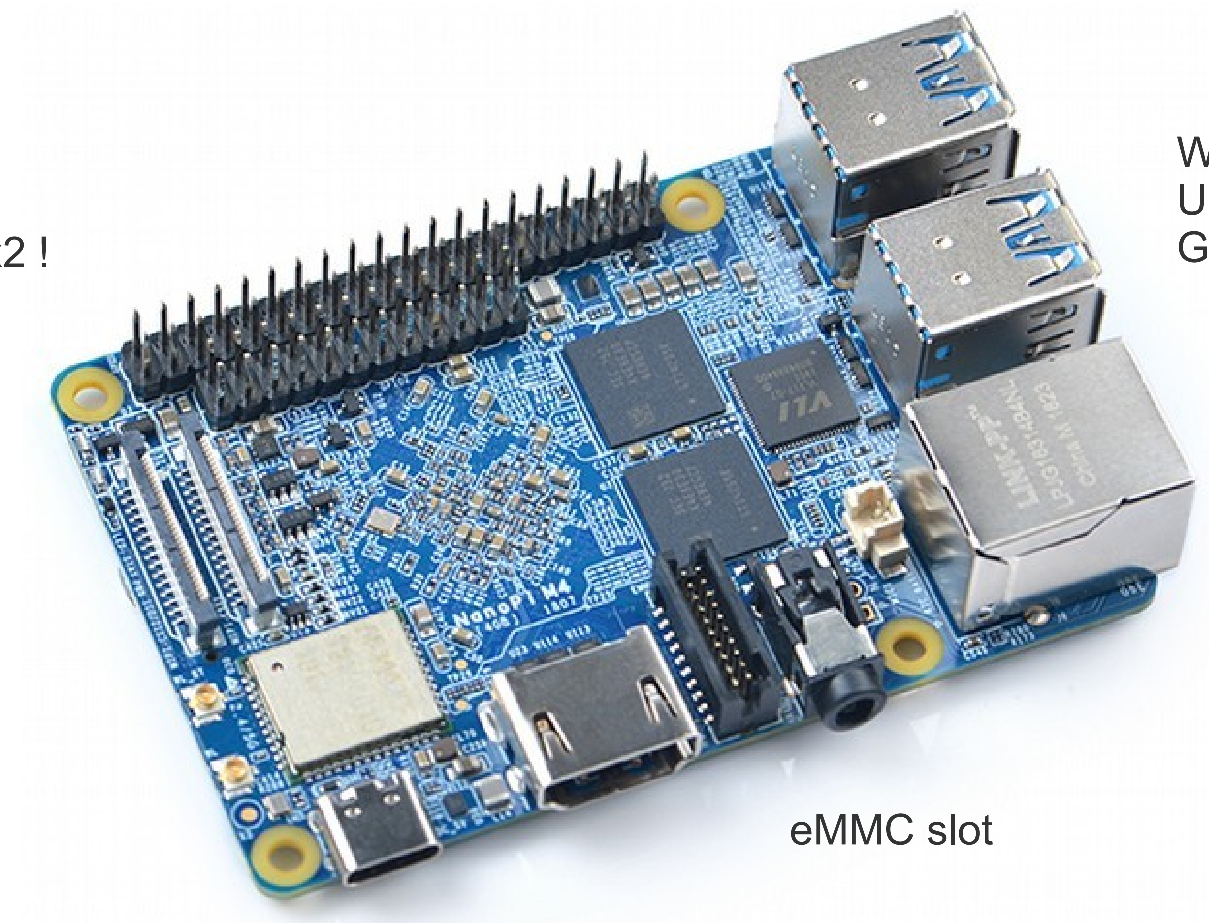
Same 40pins as Pi B+/Pi 2, 1GHz ARM11 (2835 SOC 40%faster than original Pi), 512MB Ram, µHDMI.

Board size: 65mm x 30mm

Rockchip boards

- 2017, FriendlyElec NanoPi-M4 is **RK3399** based: Dual-Cortex A72 + Quad-core A53, 2GB ~70€

PCIe x2 !



WiFi b/g/n/ac +BT 4.1
USB3
Gigabit Ethernet

eMMC slot

Credit card sized SBC@2GHz! … 100€ for the 4GB version

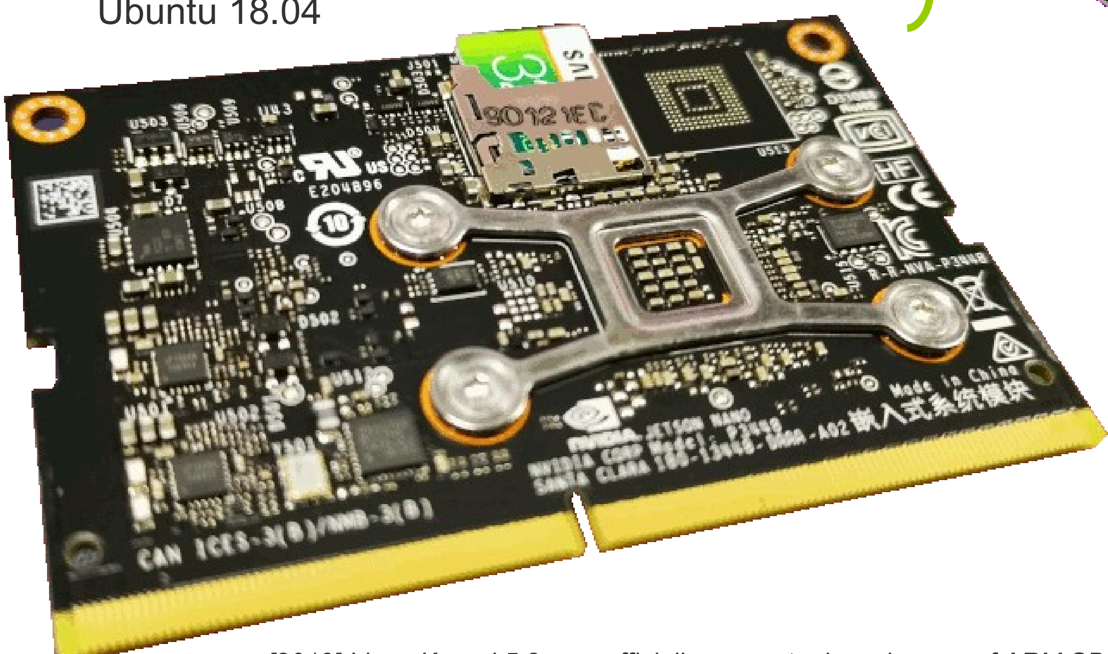
Nvidia Jetson Nano

- 2019, May ⇒ new powerful challenger in **AIoT** world for ... ~99$

[feb.20] dual CSI carrier board

- Carrier board

USB3, Gigabit Ethernet, Display Port, HDMI,
GPIO RPi fashion, PWR & RST button,
PiCamera support :)
Quad-core Cortex A57 (ARMv8@1.43GHz)
4 GB LPDDR4 @ 1.6GHz
128 cores GPU Maxwell@921MHz
Hidden M.2 Key-E socket beneath SOM
Max power consumption = **10w**
Ubuntu 18.04



- System On Module (SOM)

upcoming new SOM module with eMMC (SDCard only right now).

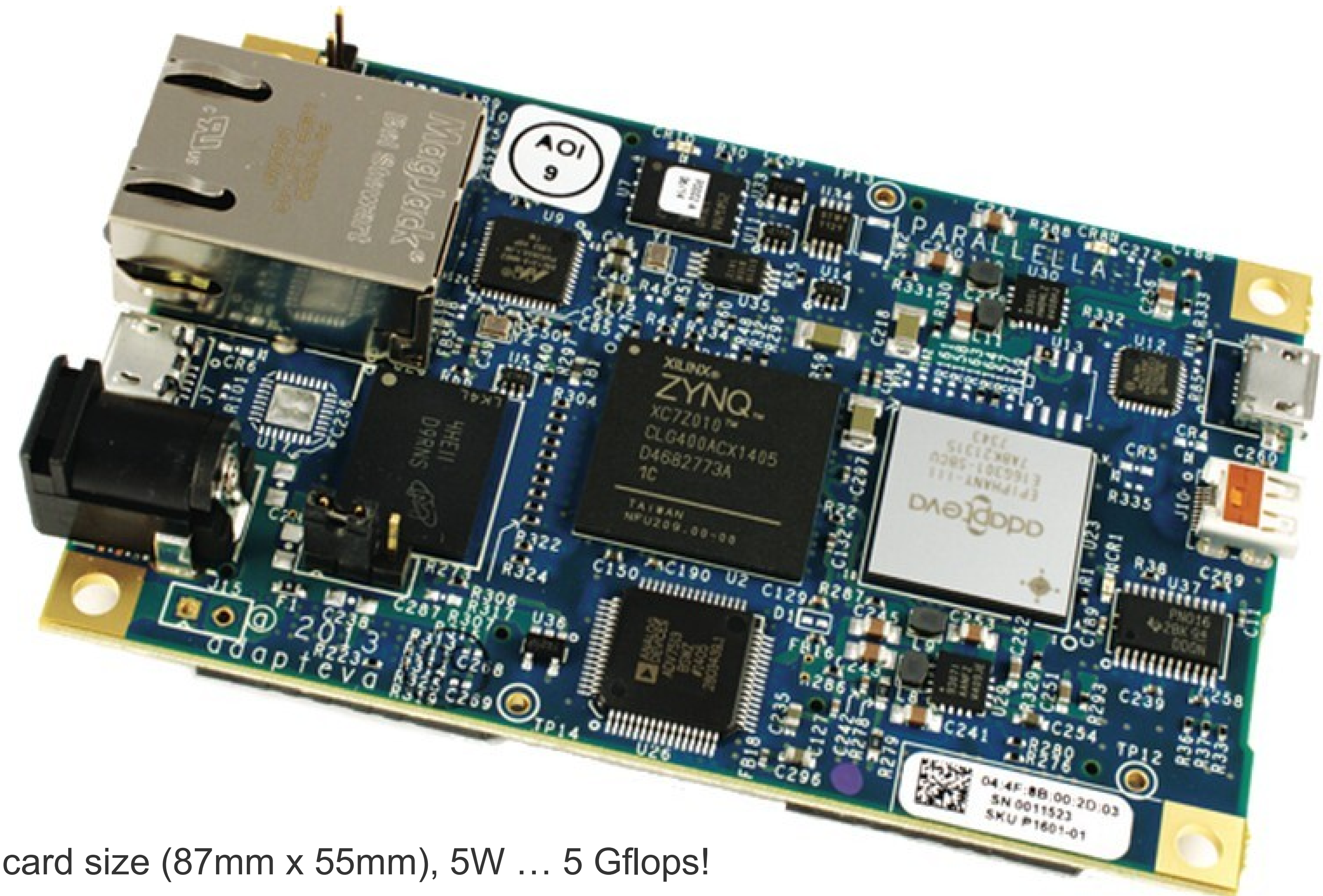
Note: optional Network boot + NFS root

[2019] Linux Kernel 5.2 now officially support a broad range of ARM SBCs like Jetson Nano, Orange Pi RK3399, NanoPi Neo4 etc etc :D

https://elinux.org/Jetson

Parallella (2015)

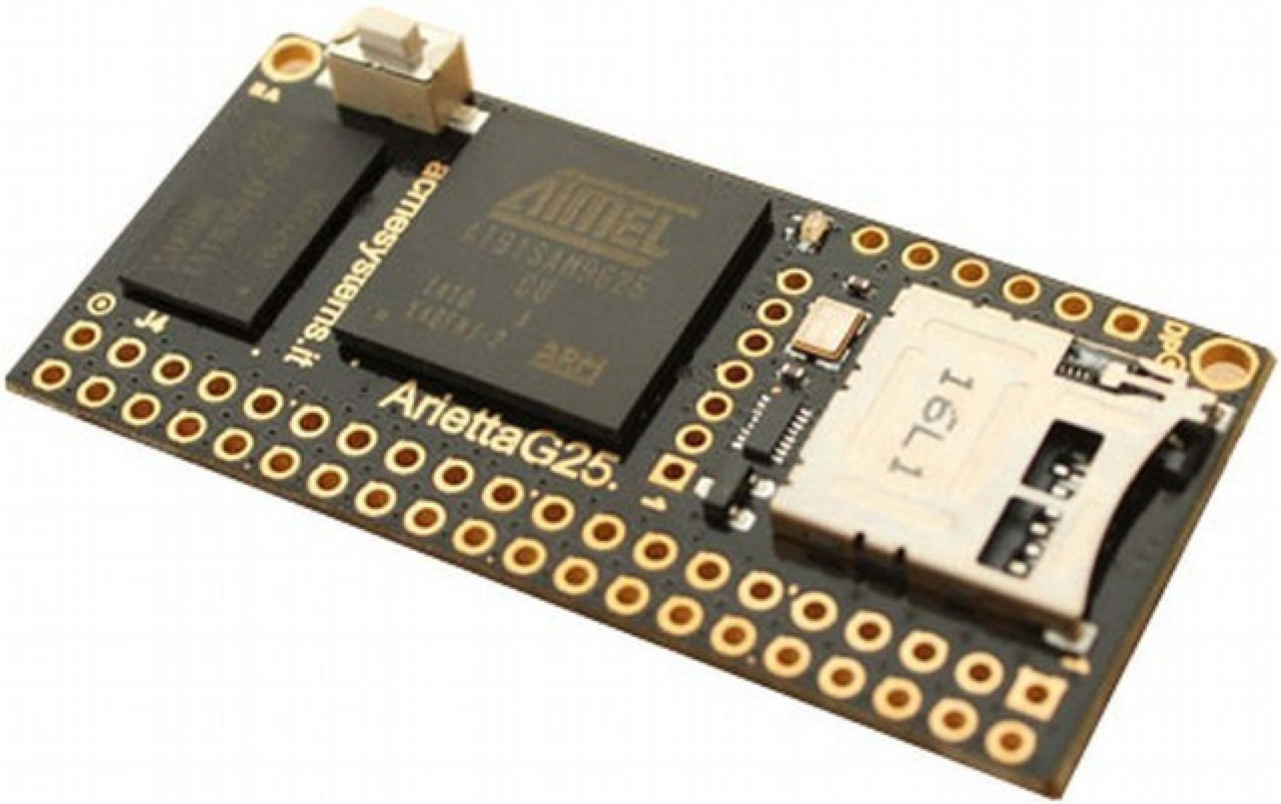
- Xilinx Zynq based (**FPGA**), dual-core ARM Cortex-A9, up-to 1GHz, 1GB DDR3 ~100€



Credit card size (87mm x 55mm), 5W … 5 Gflops!

Arietta G25 (2014)

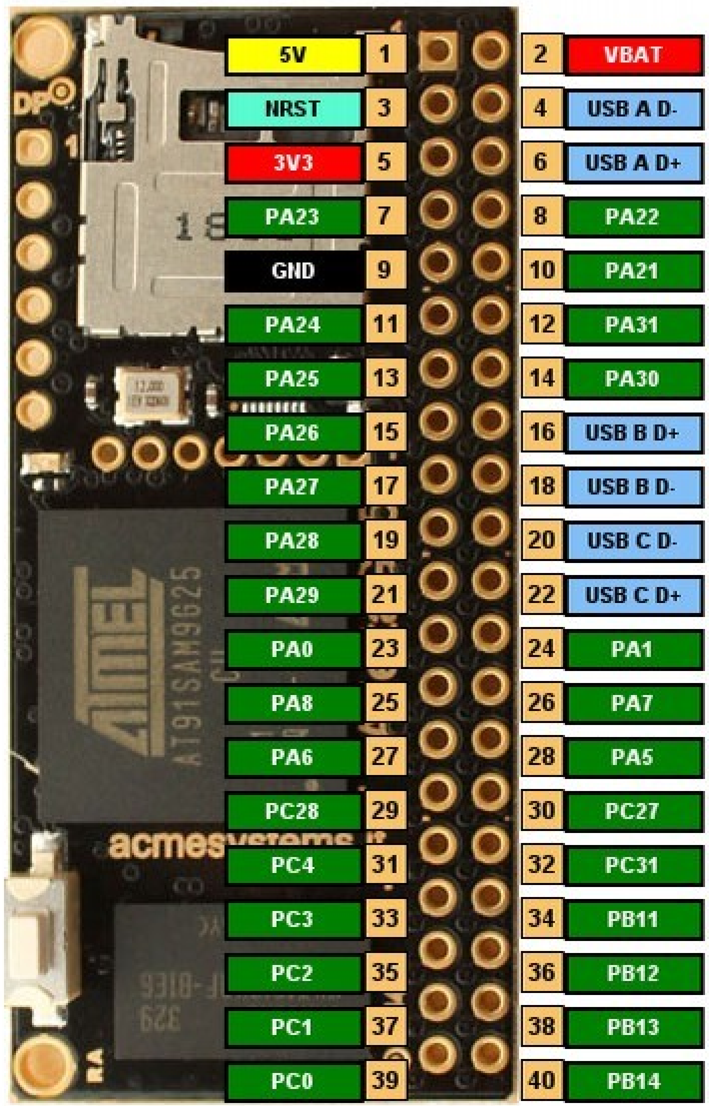
- Atmel ARM9 400MHz, 256MB ram, WiFi (optional specific USB dongle) ~36€ + usb dongle xx€

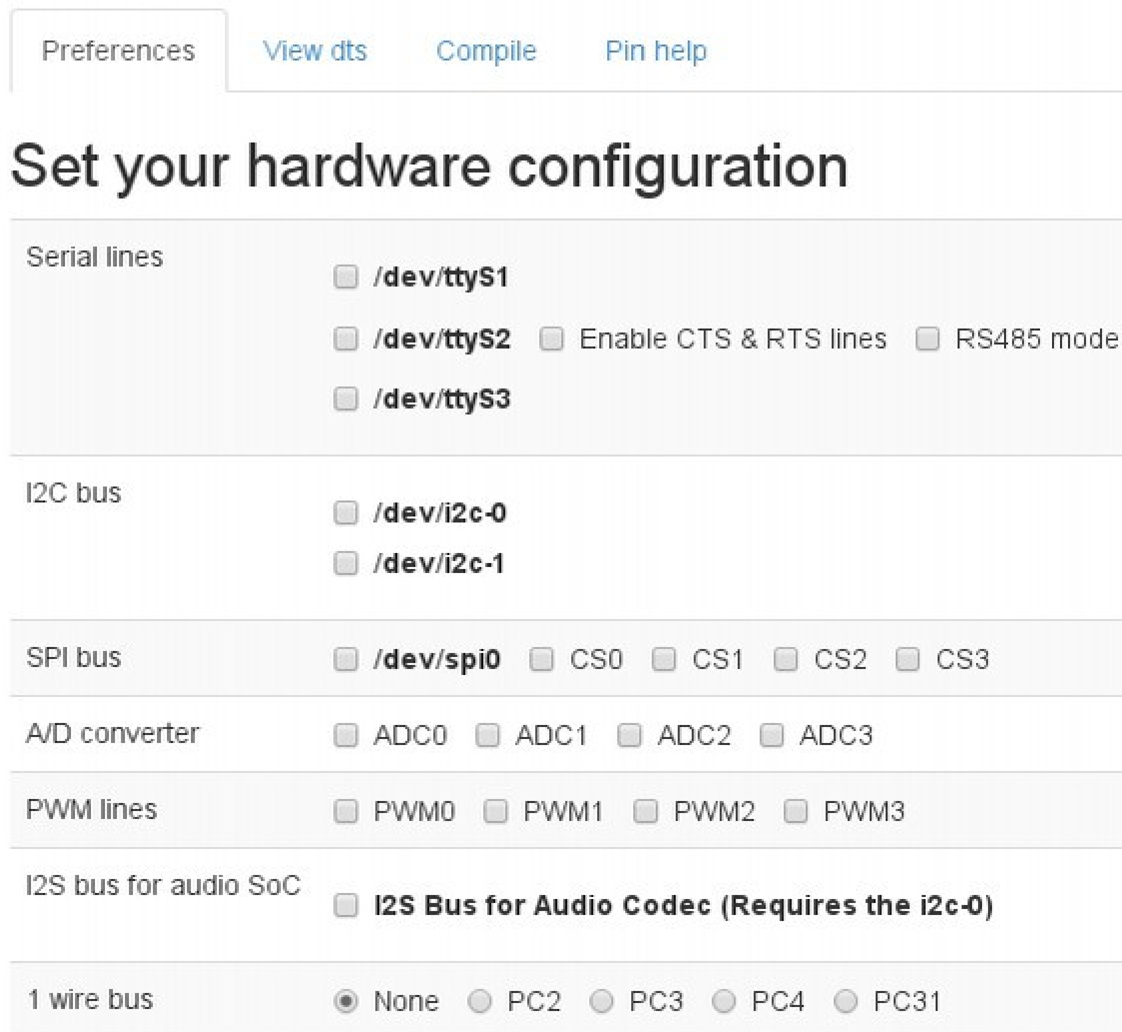


Board size: 53mm x 25mm

Arietta G25

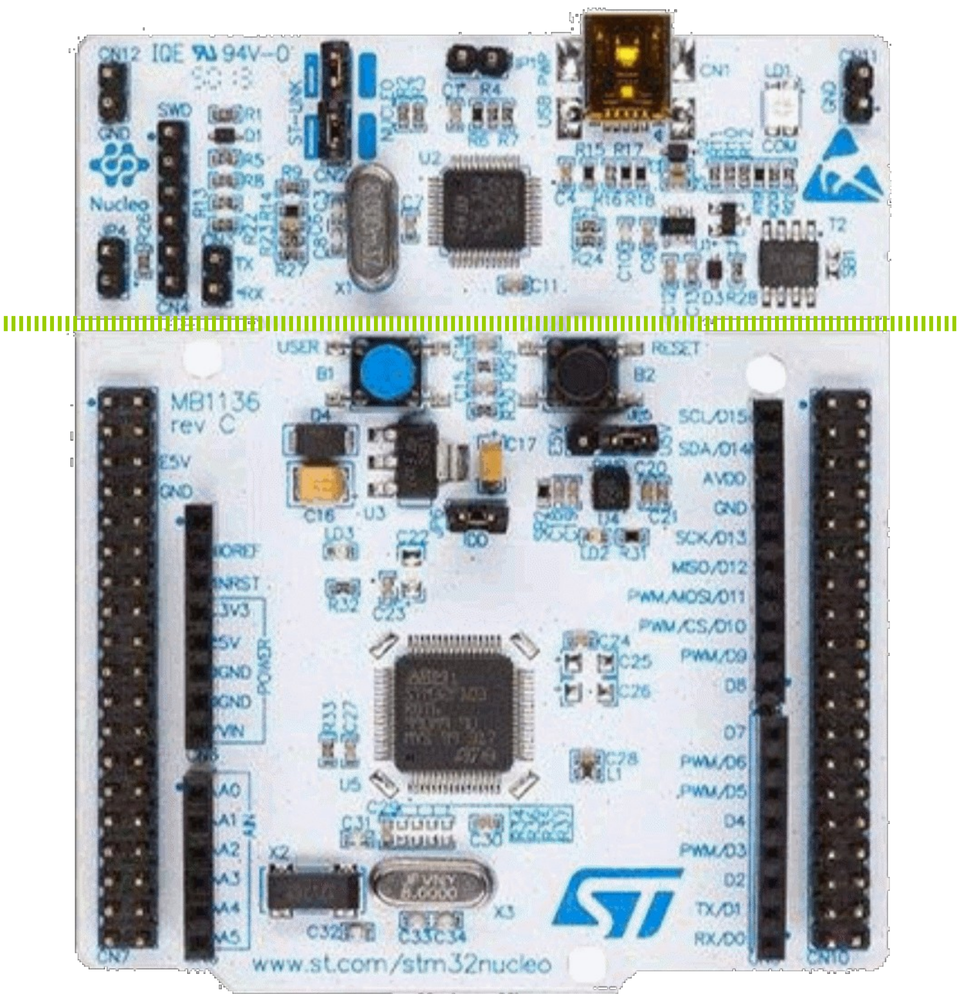
- Device tree web compiler http://dts.acmesystems.it/arietta/





Nucleo family (STmicroelectronics)

- STM32 based ARM processor boards ranging from cortex M0 to M7 ~ 13€



Nucleo-64, board size: 70mm x 82.5mm

- ARM mbed online free IDE
  https://www.mbed.com
- Arduino support: stm32duino
  https://github.com/stm32duino/Arduino_Core_STM32
- ability to snap off the programming and debugging areas of the board,
- 3 families, large range of boards

Boards	STM32	Max. pinout
Nucleo-32	Cortex M0 → M3	32
Nucleo-64	Cortex M0 → M4	64
Nucleo-144	Cortex M4 → M7	144

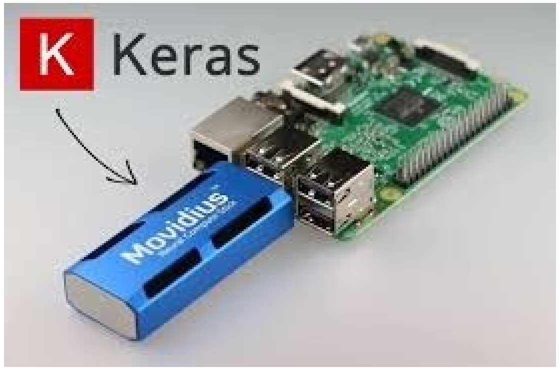
a shift towards AIoT

- [2017] Intel Neural Compute stick ~ 80€

  … bring neural network computing to SBCs!
  features an Intel Myriad2 chip

- [2020] Google TPU accelerator ~ 77€

[Sep.18] new Movidius2 based on Intel Myriad X chip, same price.



NN (Neural Netwok) processing capabilities are currently being embedded to a large range of processors either through dedicated co-processors or FPGA !

Plan

Part I – 2 SBC, communication chips ...

- Single Board Computers,
- Communication modules,
- Industrial Arduino,

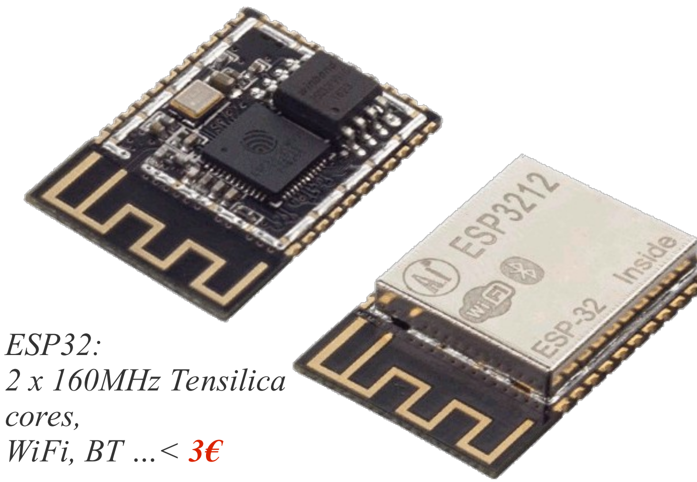
WiFi/BLE modules

- Single chip Wifi enabled | use with Arduino IDE ! … 1.75€ !!

ESP8266-12E
< 2€

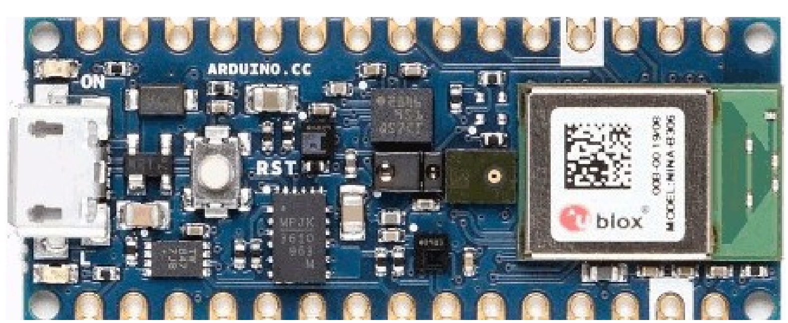
24mm x 16mm

ESP8266-12E dev. Board ~3€



ESP32:
2 x 160MHz Tensilica cores,
*WiFi, BT …< **3€***

[sep.19] new arduino



Nordic nRF52840 (Cortex M4F) @ 64MHz, 1MB Flash, 256KB RAM
BLE, Zigbee, Z-wave and Bluetooth mesh capable!

*Full sensors version … **27€***
(IMU, digital microphone, ambient parameters)

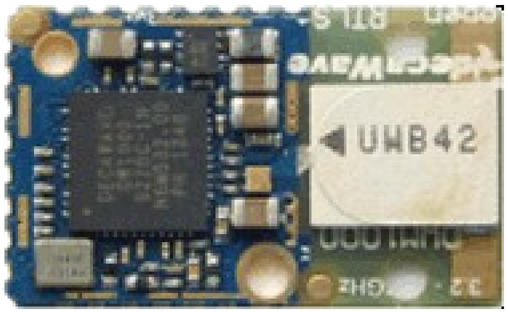
Communication modules

RFM69CW, 433/868/915MHz,
*13dBm, **AES**, SPI interface, ~3.50€*

16mm x 16mm

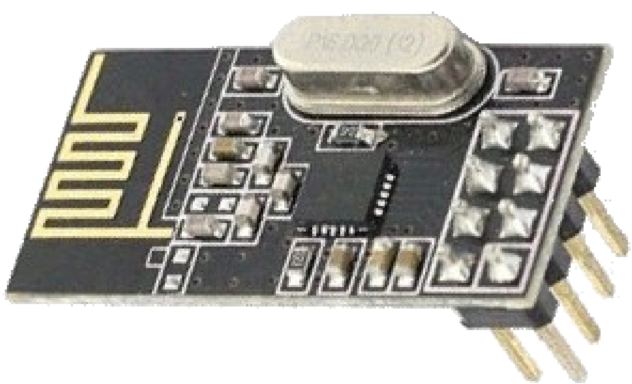
[2020] Heltec Cuve Cell LoRaWAN
module Arduino compatible, ~12€

DWM1000 3.5GHz – 6.5GHz,
comm+localization 10cm ~15€

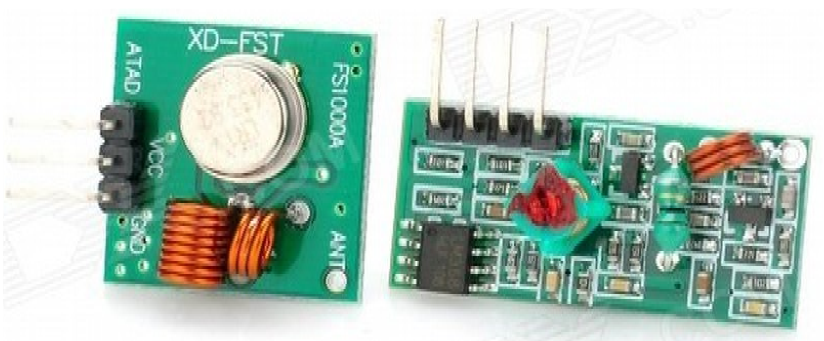


13mm x 23mm

NRF24L01+, 2.4GHz, SPI, <1€



FM433MHz emitter et receiver … 0.50€!



Plan

Part I – 2 SBC, communication chips ...

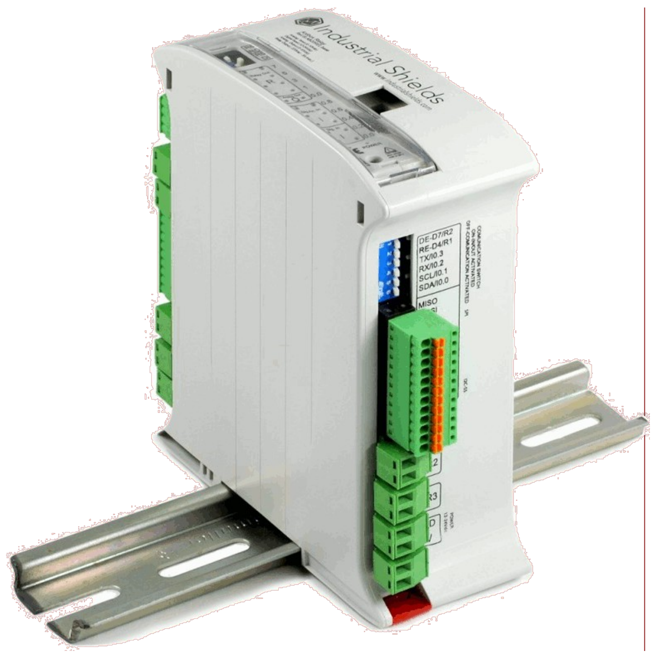
- Single Board Computers,
- Communication modules,
- Industrial modules,

Industrial Arduino

- Hardened Arduino modules

http://industrialshields.com

Ardbox-20: 10 A/D inputs, RS485, 10 A/D/PWM outputs ~135€



- *Program with open-source Arduino tool-chain (Arduino Leonardo –ATmega32u4)*
- *Used as i2c slave @ neOCampus → upto 127 slaves*

M-Duino42: 26 A/D inputs, RS485, Ethernet, 22 A/D/PWM outputs ~275€

- *Based on Atmel ATmega2560*
- *MQTT through Ethernet :)*

Industrial Arduino

- Hardened Arduino modules

http://controllino.biz

Controllino-Mini 119€ HT

Controllino-Mega 269€ HT

Industrial modules

- CGE electronics IPX800 v4

All-in-one PLC & Web server 238€

Modular Architecture (e.g power meter module addon)

Industrial modules

- SonOff modules … WiFi driven (cheap) IoT devices!

Power plug driving module with optional power consumption retrieval.

Single or dual sensitive switches one-way or two-ways + WiFi remote

DIN RAIL 4 channels relays WiFi driven
+ 868MHz transceiver (pro version)

*Note: SonOff modules are **ESP8285** based*
= ESP8266EX+1MB flash embedded

Hack all of them with ESPeasy, Tasmota or Homie firmwares!

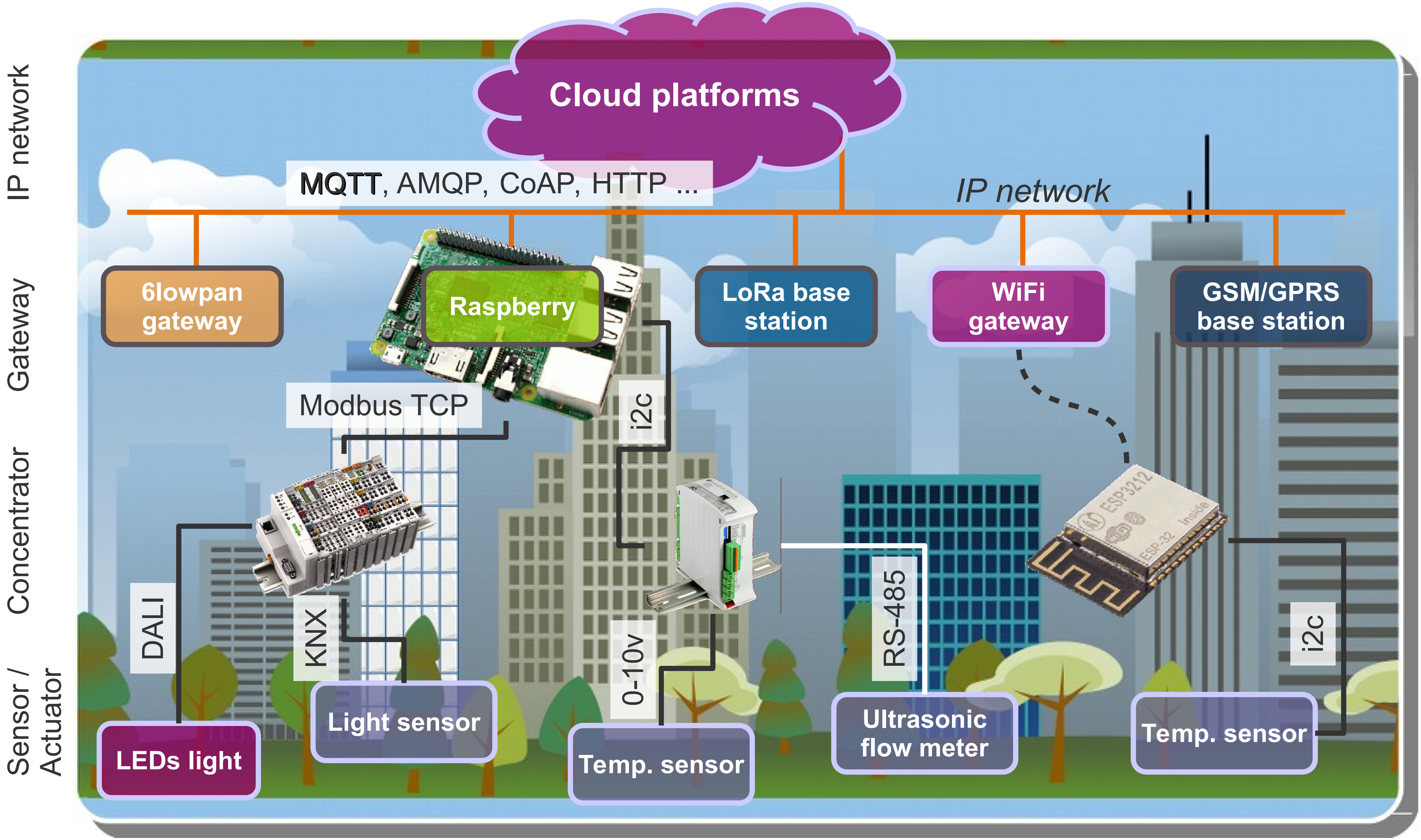
https://projetsdiy.fr/hack-sonoff-4ch-pro-firmware-mqtt-tasmota-inclusion-domoticz/

Plan

Part II – 1 IoT architecture & infrastructure protocols

- IoT framework … toward Smart cities,
- Flat-level communications | ZigBee, Z-Wave, enOCean, 6lowPan, …,
- Building-level communications | KNX, ModBus, DALI …,
- … and beyond the building | LoRaWAN, SigFox, LTE-M, ...

… toward Smart Cities



IoT stack

Services	Fully distributed agents that cooperate as services, or centralized (e.g OpenHAB scripts, IFTTT, nodered ...).
Presentation	emonCMS, OpenHAB, Domoticz (open-source solutions). Pi OSISoft, PCVue etc
Devices management	IoTivity layer, sensOCampus (neOCampus), OM2M (INSA) ...
Data management	Flow processing: nodered, IFTTT, thingspeak etc etc + Data storage: MongoDB, Apache Cassandra ...
Data acquisition @ cloud	Protocols like AMQP, MQTT (neOCampus), CoAP, HTTP, TCP ... and mostly JSON payload.
Data acquisition @ end-devices	Various kinds of wired (i2c, KNX, ...) or wireless buses (6lowpan, LoRa, WiFi, …) connected to devices with drivers
Discovery	e.g Device declares its activation to a cloud platform: QR code → local & remote setup, power on! Also part of IoTivity layer.

Device-level Cloud-level

Cloud

- Self-service storage and processing facilities featuring
  - Elasticity (CPU, RAM, disk),
  - Security (SSL, Kerberos, ...),
  - Availability,
  - Paid per-use.

http://cloudmip.univ-tlse3.fr

- CloudMIP ...
  - OpenStack (widely used open-source solution),
  - 280 core with ~2TB RAM,
  - 24/7 availability,
  - Up to 45TB persistent storage,
  - Public IPs.
- ... is part of France-Grilles → 10 OpenStack sites, > 1000 CPU,

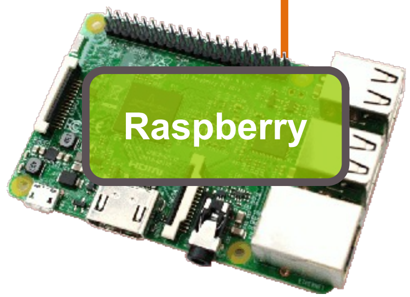
Plan

Part II – 1 IoT architecture & infrastructure protocols

- IoT framework … toward Smart cities,
- Flat-level communications | ZigBee, Z-Wave, enOCean, 6lowPan, …,
- Building-level communications | KNX, ModBus, DALI …,
- … and beyond the building | LoRaWAN, SigFox, LTE-M ...

Comm. @ flat-level

- Sensors / Actuators (devices) communicate with a base station (gateway) via:
  - Wired network: CPL, Ethernet,
  - Short range Wireless network: WiFi, BLE, 868/433MHz, enOCean,
    and meshed wireless: 6lowPAN, ZigBee, Z-Wave.
    [Q1-19] Bluetooth mesh network; BT5 extends range to 40m indoor



- Possible gateways (base station) are:
  - Stand-alone concentrator connected to an Internet Box (+ uPnP protocol),
  - Neighbour's concentrator (meshed network),
  - Integrated / plug-able communication device directly @ Internet box-level
    (e.g Freebox Delta featuring SigFox connectivity)

Flat-level comm.: synthesis

Name	Frequency	Payload	Density	Notes
enOCean	868.3MHz	14bytes app.payload 125kpbs		ASK, FSK modulation Energy Harvesting (piezo, peltier, solar)
ZigBee *ZigBee IP*	2.4GHz 868MHz	101bytes app. payload 250kbps (2.4GHz) 20kbps (868MHz)	16bits PAN ID (64bits now), > 1000 end-points	FHSS modulation. Meshed network. IP version provides ipv6 connectivity.
6lowPAN	2.4GHz, sub-GHz	Ipv6 MTU=1280bytes 802.15.4 MTU=127bytes 81bytes app. payload		Thread (ipv6 stack) OpenThread
Z-Wave	868.42MHz	46bytes app. payload 40kpbs	232 devices	Meshed network, integrated within some end-users appliances. Specs available since sep.16!
FSK modules	868MHz 433MHz	66 bytes, upto 38.4kbps	250 devices per channel, 250 channels	Listen to single channel, RFM69CW < 3€
WiFi modules	2.4GHz			Low-cost programmable devices like ESP8266
Bluetooth 5	2.4GHz	< 255 bytes		BLE, eBeacon, long-range (BT5), AoA (BT5.1) 40m range indoor, 120m outdoor Higher speed than BT4.2 Dual audio

Meshed networks: 6lowPAN, ZigBee, Z-Wave, BT mesh → extended coverage but less battery friendly

Plan

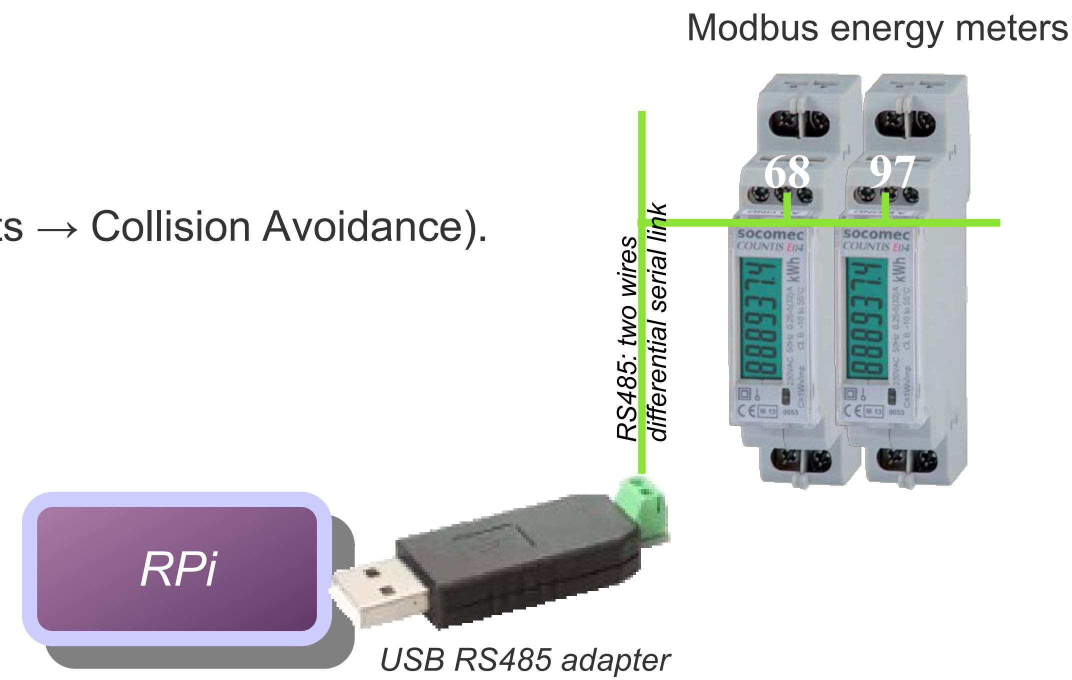
Part II – 1 IoT architecture & infrastructure protocols

- IoT framework … toward Smart cities,
- Flat-level communications | ZigBee, Z-Wave, enOCean, 6lowPan, …,
- Building-level communications | KNX, ModBus, DALI …,
- … and beyond the building | LoRaWAN, SigFox, LTE-M ...

Comm. @ building-level

- Building automation will make use of PLCs as gateways for industrial buses like:
  - KNX (LON successor),
  - DALI for lighting management (replace old 0-10v dimmers),
  - ModBUS, simple and well established protocol (TCP, RS-232, RS-485) that will read/write remote devices’ memories.

- Some specialized buses:
  - CAN and LIN (automotive),
  - Ethernet Powerlink (timeslots → Collision Avoidance).



Comm. @ building-level

Name	Wires	Speed	Range	Notes
KNX	2 wires, differential	9600bits/s	1000m and up to 4x1000 (repeaters)	No free-of-charge programming software. 120Ohm termination
ModBUS RS485	2 wires differential	9600 and 19200bauds (mandatory). Higher speeds available.	1200m	Differential signaling. 120Ohm termination resistor. Usually cat.5 cable
DALI	2 wires differential	1200bauds	300m (more with repeaters)	Self-powered network. 18v typ. power supply. Upto 64 ballasts per segment
ModBUS TCP	Ethernet cable			IP connectivity
CAN	2 wires, differential	1 Mbits/s	40m (1 Mbits/s), 500m (125 kbits/s) 6km (10 kbits/s)	Multi-master, CSMA/CD, 120 ohm termination
Ethernet powerlink	*Cat. 5(+) UTP full duplex*	*125MHz*	*Repeater every 60m*	4wires = 100mbps 8 wires = 1000mbps Hubs based

M-Bus used for power consumption retrieval is RS-485 based. Up to 250 nodes per bus.
Wireless version (wM-Bus) is superseded by LoRa network.

Plan

Part II – 1 IoT architecture & infrastructure protocols

- IoT framework … toward Smart cities,
- Flat-level communications | ZigBee, Z-Wave, enOCean, 6lowPan, …,
- Building-level communications | KNX, ModBus, DALI …,
- … and beyond the building | LoRaWAN, SigFox, LTE-M ...

Beyond the building ...

- LoRa network
  - is a **LPWAN** (Low Power Wide Area Network) like SigFox,
  - technology acquired by Semtech in 2012,
  - LoRA is mostly a layer 1 (OSI stack) ranging from 169MHz to 1GHz,
  - Mostly known implementation focus on the 433/868MHz (ISM) frequency,
  - Chirp spread spectrum (CSS) modulation (FHSS for ZibBee and BT),
  - enables point to multi-points networks,
  - LoRa modules available (like RFM95W with a 20dBm PA),
  - LoRaWAN is (OSI) uppers layers,
  - Bouygues, Orange and others have deployed LoRaWAN by the end of 2017.

TTN indoor gateway #75€HT

[2020] Helctec Cube Cell, LoRaWAN arduino compatible board ~12€

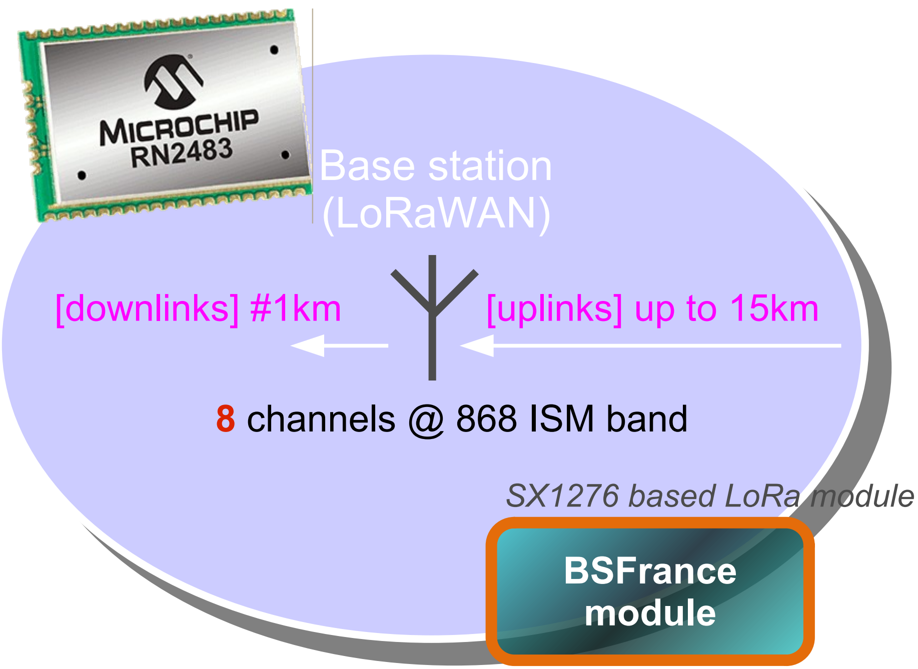
LoRa vs LoRaWAN ?

OSI – Transport layer	LoRaWAN	UDP/TCP
OSI – Network layer		IP
OSI – DataLink Layer		MAC + LLC
OSI – Physical Layer	LoRa	Ethernet

LoRa network (cont'd)

https://www.lora-alliance.org

- Max. 50kbps,
- App. payload up to 222 bytes,
- 125 to 250kHz bandwidth,
- 1% & 10% duty-cycle @ 868MHz,
- LoRaWAN server @ base station,
- LoRaWAN manages data-rate AND power of each end-device, (Adaptative Data Rate –ADR)
- Allow Private Network



Université Toulouse 3 – neOCampus

Three LoRaWan gateways deployed to manage the end-devices both used in teaching and research projects

BSFrance LoRaM3-D L151 / STM32 + OLED + LiPo + SX1276

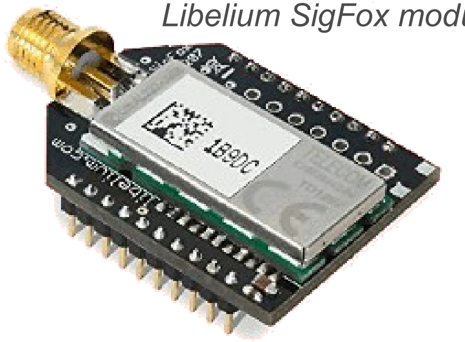
Beyond the building ...

- SigFox network
  - is a **LPWAN** (Low Power Wide Area Network), established before LoRa,
  - mostly a layer 1 (OSI stack) available at 868MHz and 915MHz (ISM bands),
  - (ultra) narrowband BPSK modulation, 10 – 40km range, 100Hz bandwidth,
  - 140 message per-object per-day, 12bytes app. payload, 100bps,
  - no downlink message (as of 2020),
  - SigFox module (# 50€) comes with an annual subscription fee (15€),
  - SigFox end-nodes can only communicate with SigFox's bases stations,
  - **NO PRIVATE** network.

[2016] Université Toulouse 3 – SGE

Deployment of a SigFox based solution to retrieve water consumption from buried counters.

Libelium SigFox module



Beyond the building ...

- NB-IoT, LTE-M and future 5G IoT

… are the 3GPP consortium answers to LPWAN networks.

• **NB-IoT** (LTE Cat NB1) QPSK modulation, NOT part of the “*Long Term Evolution*” (i.e 4G), unlimited msgs, 375kbps, half-duplex, long battery life, fixed devices.

• **LTE-M** (LTE Cat M1) is a 4G extension for IoT networks, intended for moving devices.

• Future **5G IoT** (3.4 – 3.8GHz) … will enable V2V (Vehicle to Vehicle) and V2X with high throughput and low latency. First modules available from uBlox and Orange.

[nov.19] uBlox SARA N2, NB-IOT module, 12€ for a 100 units minimum order. RX: 46mA, TX: 74 - 220mA

[apr.18] First LTE-M modem from Link-Labs
Works with Verizon network
249$ dev. Kit

Note: up to 500mA TX and 330mA RX !

Beyond the building ...

Name	Freq.	Range	Power consumption	Notes
LoRa	868MHz	1- 15km -148dBm sensitivity	TX: 40mA @ 14dBm RX: 14.2mA	Microchip RN2483 (~15€), programmable module. +10y battery life expected
SiGFox	868MHz	10 - 40km -126dBm sensitivity	TX: 37mA @ 10dBm RX: 13/16mA	Module (~50€) + 15€/y
NB-IOT (LTE-Cat NB1)	Licensed 800MHz		TX: 100mA @ 13dBm* RX: 46mA **72mA(-40dBm), 220mA (23dBm)*	Low data usage over long period of time
GSM/GPRS	850MHz 900MHz 1.8GHz 1.9GHz		Continuous: 500mA Pulse (TX ?): 2000mA 1.5mA sleep mode	GSM/GPRS (SIM900) module #50€
ESP8266	2.4GHz	< 100m	TX: 140mA	Popular WiFi module used with USB power adapter in neOCampus.

Note: EU 868 ISM duty-cycle is 1% for end-nodes
e.g: 0,5s TX → wait for 49,5s before new TX

OSI layers synthesis

	6lowPAN	WiFi	LoRaWAN
Application	Nest Weave	HTTP, CoAP, MQTT	
Transport	THREAD Networking Protocol	TCP / UDP	LoRaWAN
Network	ipv6	Ipv4, ipv6	
Data link	6lowPAN IEEE 802.15.4 MAC	IEEE 802.11 MAC / PHY	
Physique	IEEE 802.15.4		IEEE 802.15.4g LoRa

IEEE 802.15.4: 250kbits/s with 127bytes payload

Note: OSI layers 5 (session) and 6 (presentation) not shown.

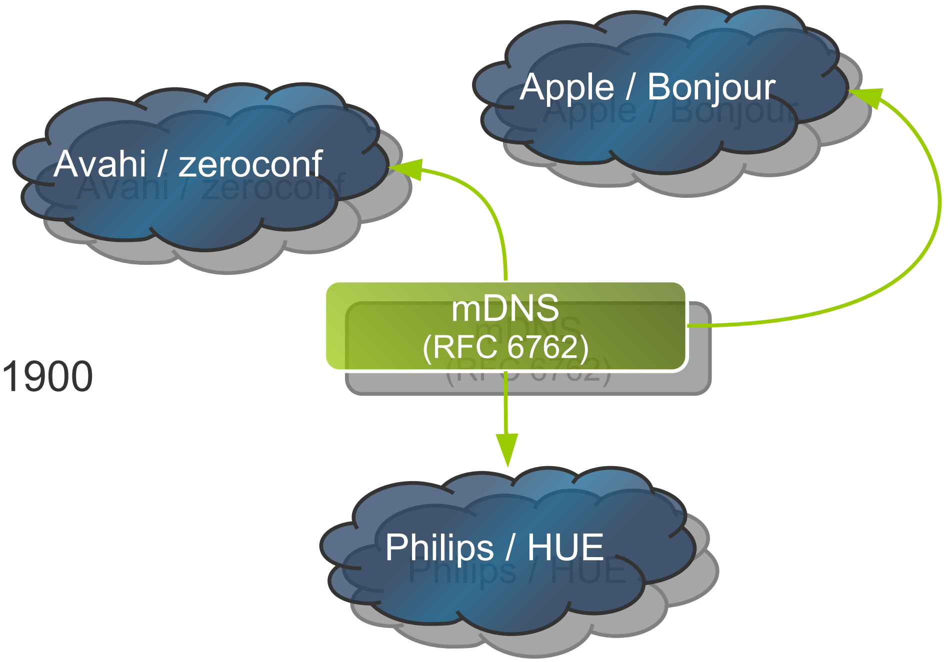
Plan

Part II – 2 Data protocols, services and applications

- Discovery,
- Data acquisition,
- Flow processing,
- Data management,
- Presentation,
- Ambient Applications.

Discovery

- mDNS
  - Distributed DNS for **local** network (i.e same vlan),
  - Multicast (ipv4) 224.0.0.251, UDP port 5353,
  - Multicast ask for IP of <*hostname*>
  - Response from device using multicast (all nodes then cache the IP addresse associated),
  - e.g in your browser https://esp8266.local



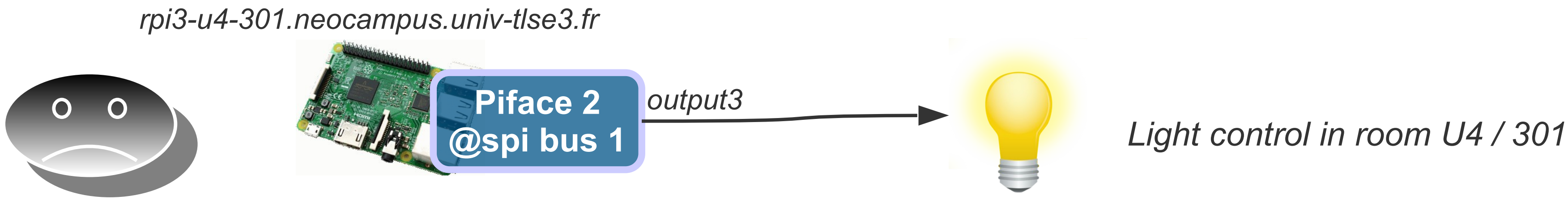
- UPnP
  - Multiple profiles (IGD, AV ...),
  - Multicast (ipv4) 239.255.255.250, UDP port 1900

- extension to Service Discovery
  - mDNS-SD,
  - UPnP SSDP (Simple Service Discovery Protocol).

Data acquisition

- We want to give users / applications access to **useful data** without hassle about networks, sensors technology or underlying embedded systems.

✖ High level of hardware details



✔ Useful data

Data acquisition

- MQTT, almost the ‘de facto’ IoT protocol ;)
  - Network addressing independent,
  - Works behind firewalls,
  - publish / subscribe paradigm (no more pooling),
  - embedded security (login / passwd) + TLS,
  - Support for WebSockets,
  - Bindings for almost all languages,
  - Paho-mqtt (python client), Mosquitto (C written client & server),
  - MQTT bridges for multiples brokers setup,
  - Topics based real-time exchanges,
  - Topics are arbitrary tokens separated with ‘/’
  - Payload agnostic (mainly json).

MQTT data trade-off:
topics vs payload content.

You may also have a look to AMQP (e.g RabbitMQ) or CoAP (Constrained Application Protocol)
Adafruit provides a free MQTT broker with a data visualisation GUI on a per-user basis.

Data acquisition

- Topics segmentation in neOCampus

u4 / campusfab / shutter / command

Base | type | [optional] command

Base : defined at <u>*device*</u> registration time according to location

e.g u4 / 300 or bu / hall …

Type : kind of sensor / actuator (<u>*module*</u>) defined by sensOCampus or automagically detected

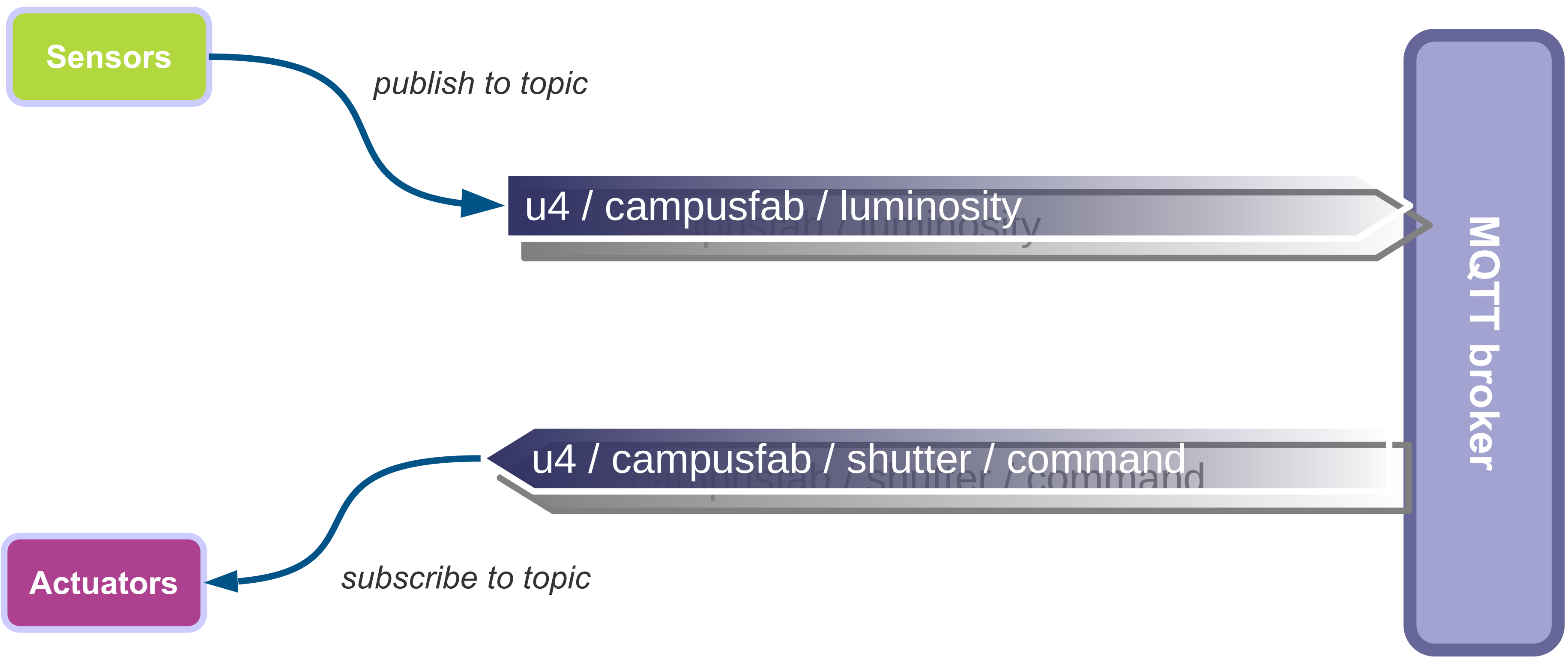
e.g shutter, luminosity, temperature, sound, lighting …

Command : to send orders to a sensor / actuator (*module*)

e.g orders to shutter like UP, STOP, DOWN

Data acquisition

- Real-time data exchange through topics



Since actuators initiate a TCP connection to the broker, they can be sent data back from the broker even when they are located behind a firewall (e.g Internet box).

Data acquisition

- MQTT payloads @ neOCampus

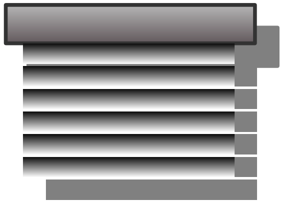
✔ Sending order to a shutter (with proper mqtt login / passwd)

```
order: "up"
dest: "all"
      or "<shutter_ID>"
```

Json frame as mqtt payload

u4 / campusfab / shutter / command

Shutter



✔ … then shutter publish its status back

```
order: "idle"
unitID: "<shutter_ID>"
status: "open"
```

Json frame as mqtt payload

u4 / campusfab / shutter

One caveats is that you can't send an order to a single module (shutter), hence the `dest` *field.*

Data acquisition

- Wildcards for multi-topics subscribing

u4 / # / temperature

multi-level subscribing (e.g u4 / campusfab / temperature, u4 / hall / box1 / temperature)

u4 / + / temperature

single-level subscribing (e.g u4 / campusfab / temperature, u4 / 301 / temperature)

- Single topic publishing

MQTT specifications does not allow to publish
to topics containing wildcards.

The `multiple` *method enables you to publish multiple data to multiple topics in a one-shot way.*

Flow processing

- Node-RED

  IBM open-source technology preview,

  NodeJS written,

  Will glue IoT elements together (e.g connecting IFTTT to MQTT, Amazon Echo to Domoticz etc etc).

  Open-source,

  Large range of inputs / outputs (MQTT, HTTP, TCP, UDP, LoRa),

  … lots of functions (nodes) for storage (MongoDB, files), javascript user code, email, social networks …,

  Custom dashboards :) → https://neocampus.univ-tlse3.fr/nodered/ui

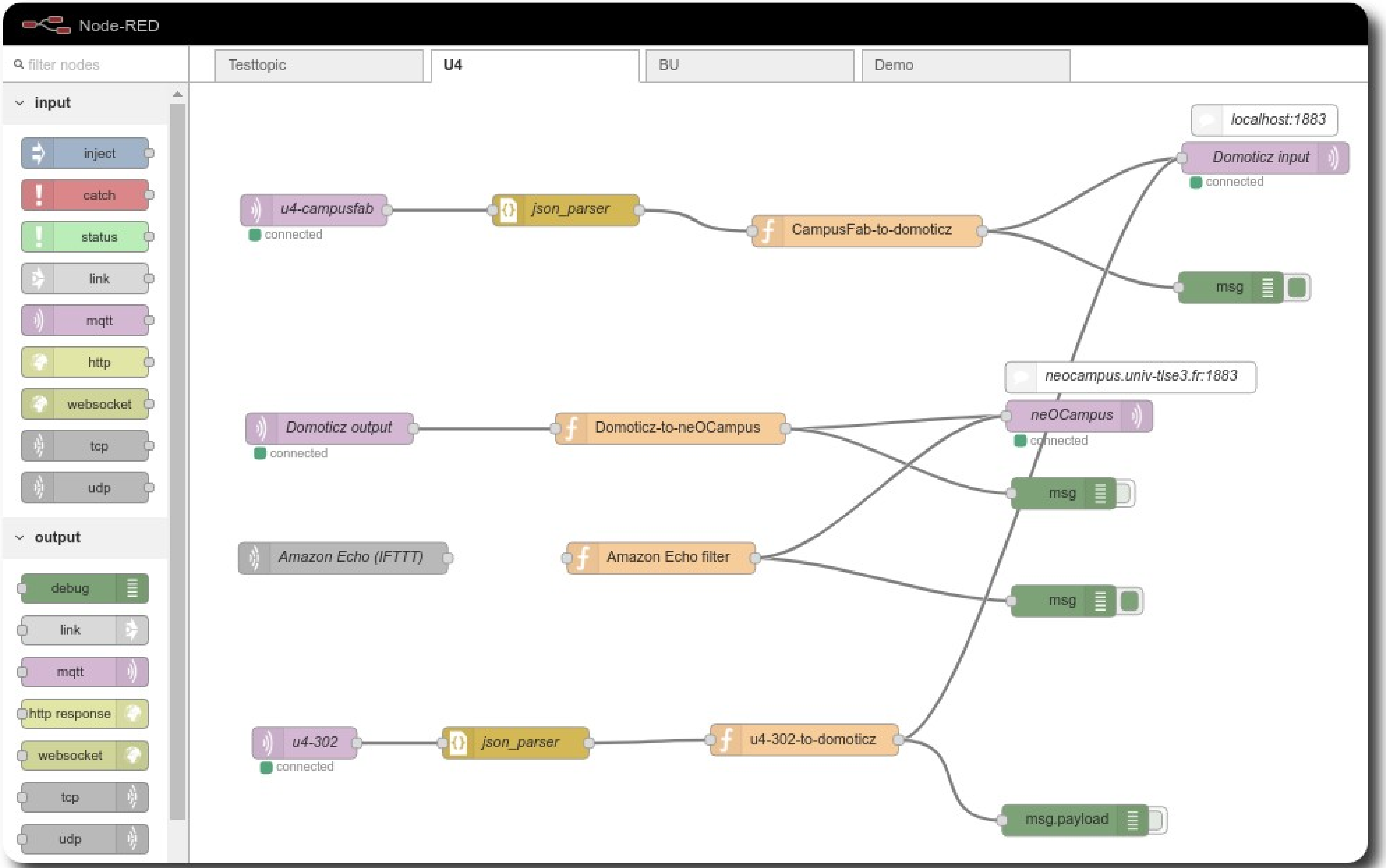
  Docker available

  Server-side execution of scripts (!),

  No per-user flows.

You may also have a look to IFTTT (i.e If This Then That)

Flow processing



Data storage

- MongoDB – documents oriented database

  database → multiple collections → documents within collections,

  No SQL, data agnostic, queries may parse the entire collection !! (possible to create indexes to speed-up queries)

  Not efficient for storage of sensors' values.

Docker volume mount point
/var/lib/postgresql/data @ Postgres
/data/db @ MongoDB
/var/lib/influxdb @ InfluxDB

- InfluxDB – time series database

  Efficient for storage of sensors' values,

  Able to process request (e.g mean value of a sensor across a timespan),

  Rest API for data ingestion & extraction, plug-ins architecture, can handle complex SQL requests, connexion to Apache Zeppelin :)

- Apache HBase – large scale distributed (key,values) pairs database

  Hadoop back-end (distribution & coherency),

  Intended for large scale deployment (e.g data lake across multiple sites),

  Connexion to Apache Zeppelin.

Presentation

- Grafana

From The Things Network

- Domoticz

    NodeJS written,

    Container available.

Open-source,

Simple, efficient and very popular,

Lua programming mode available,

Alarm panel integrated,

Difficult to map sensors / actuators (idx based),

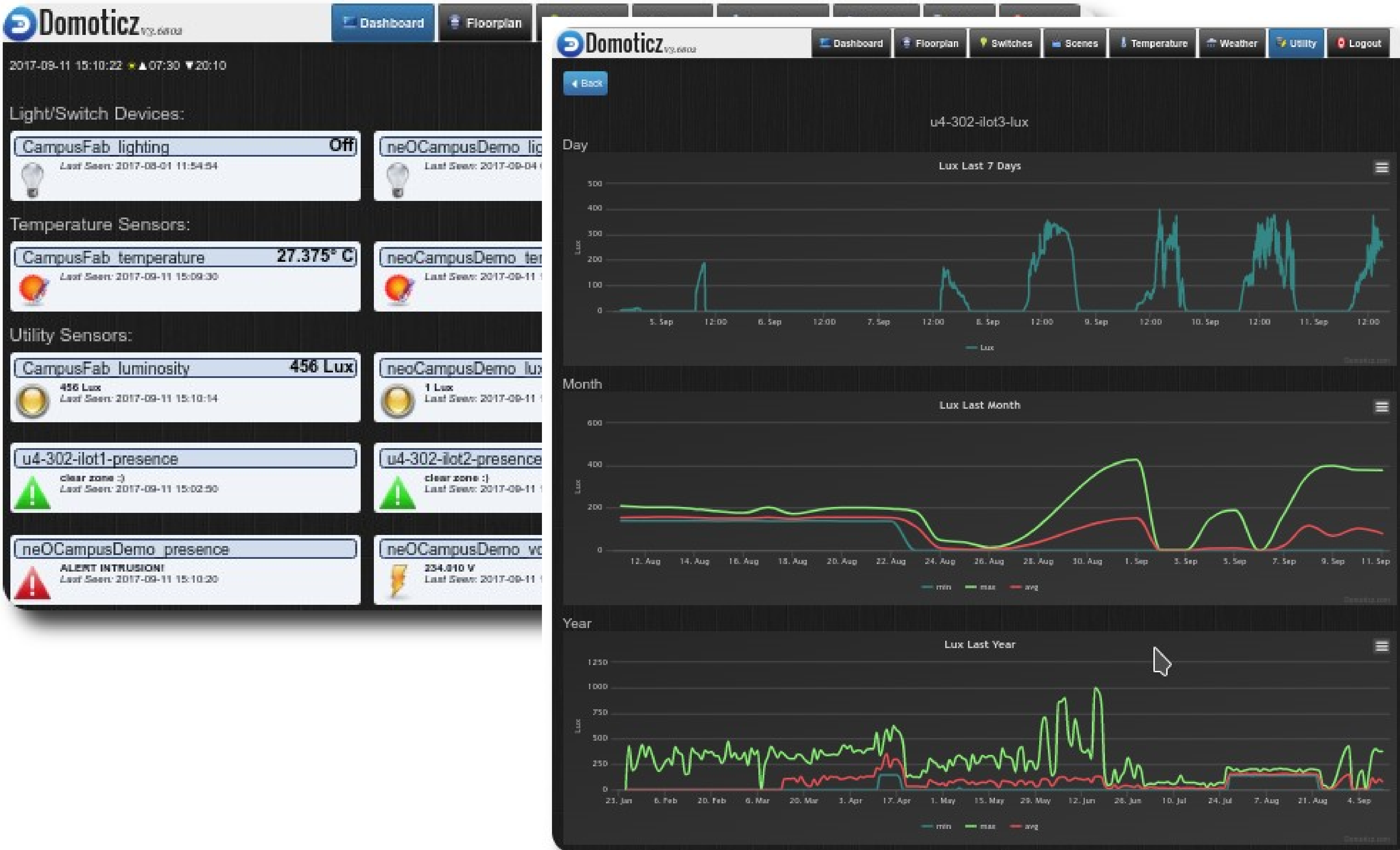
Fails to upgrade,

SBC oriented, best to use for home automation only.

- Apache Zeppelin

You may also have a look to EmonCMS, OpenHAB and Grafana.

Presentation



Ambient applications

- Ambient intelligence ...

  Ambient systems are not a collection of technological objects that communicate but a multidisciplinary, user-centred, approach.

  Ambient intelligence provides a service through cooperation between agents spread across the network.

- … applied to Edge/FOG computing

  Deployment of agents within the network nodes and/or end-devices …

  … but how to ensure in-between applications insulation ?
  ==> Linux containers !!

Plan

Part III - The neOCampus use case

- neOCampus | the big picture,
- Open-source Industrial IO | RPi + Ardbox solution,
- Access control system | accessOCampus,
- neOSensor @ BU,
- Power metering @ neOCampus,
- LoRaWAN federated gateways.

In the meanwhile, have a look to https://neocampus.univ-tlse3.fr

Plan

Annexe-A | Electronics for ~~dummies~~ all :)

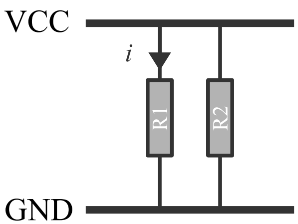
- Basics,
- CMOS transistors,
- Bipolar transistors,
- Sink / Source output modes,
- Operational Amplifier,
- Motors,
- Wavelengths,
- PCB software.

https://www.electronics-tutorials.ws/

https://doc.lagout.org/electronics/Electronics%20for%20Dummies%20%282005%29.pdf

Ohm's law and friends

Resistors

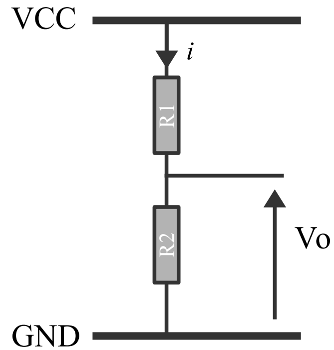


$$i = \frac{VCC}{R1}$$

$$R = \frac{R1 \times R2}{R1 + R2}$$

If R1 == R2
==> Req = R/2

R-based adaptation levels



$$i = \frac{VCC}{R1 + R2}$$

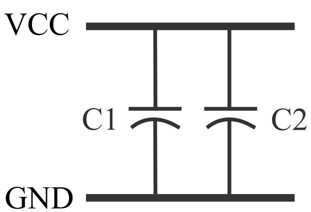
$$Vo = R2 \times i$$

$$Vo = \frac{R2 \times VCC}{R1 + R2}$$

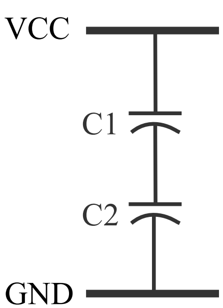
Q *Arduino features VCC = 5v but we need to drive a 3.3v input of a Raspberry Pi*

- *Compute R1 and R2 values*
- *If R1 == R2, compute Vo*

Capacitors

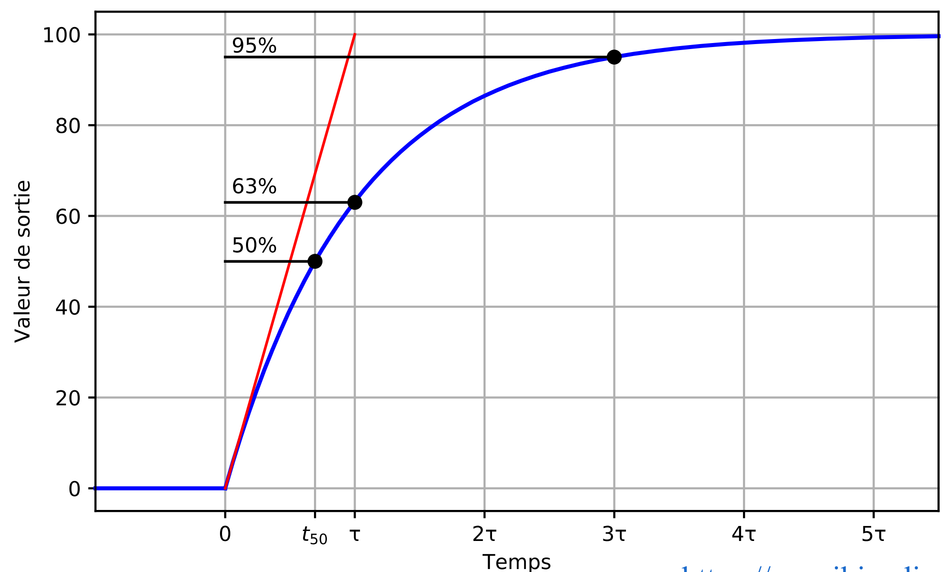


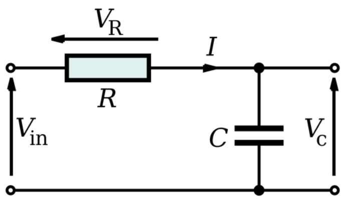
$$C = C1 + C2$$



$$C = \frac{C1 \times C2}{C1 + C2}$$

RC time constant





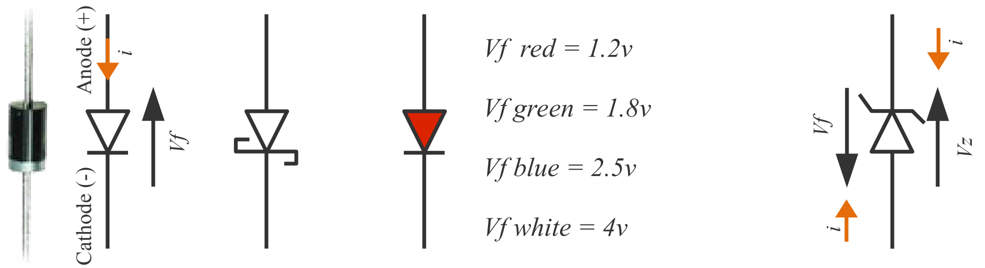
$$\tau = RC$$

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

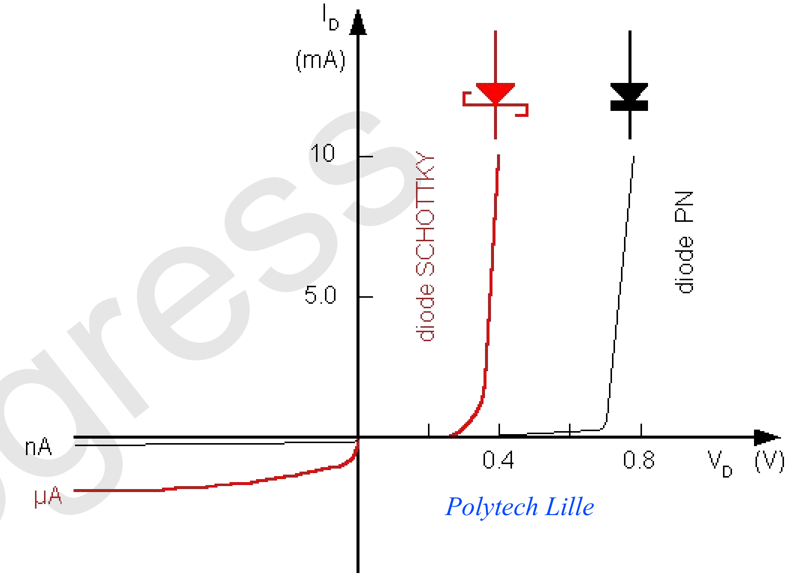
Ohm’s law and friends

- Diodes … almost unidirectional device

Common diode — *Schottky* — *Light Emitting Diodes (**LED**)* — *Zener (bi-directional)*

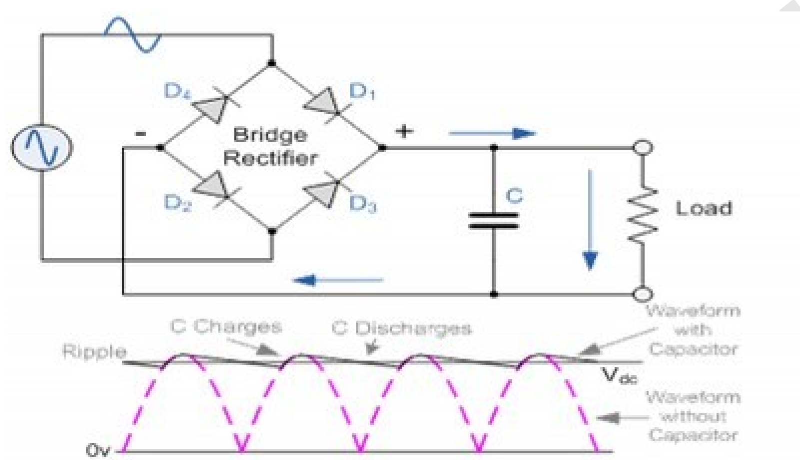


Vf = 0.6v — *Vf = 0.3v* — *Different Vf according to LED’s colour.* — *Large range of Vz*



Polytech Lille

- Bridge rectifier



- Self

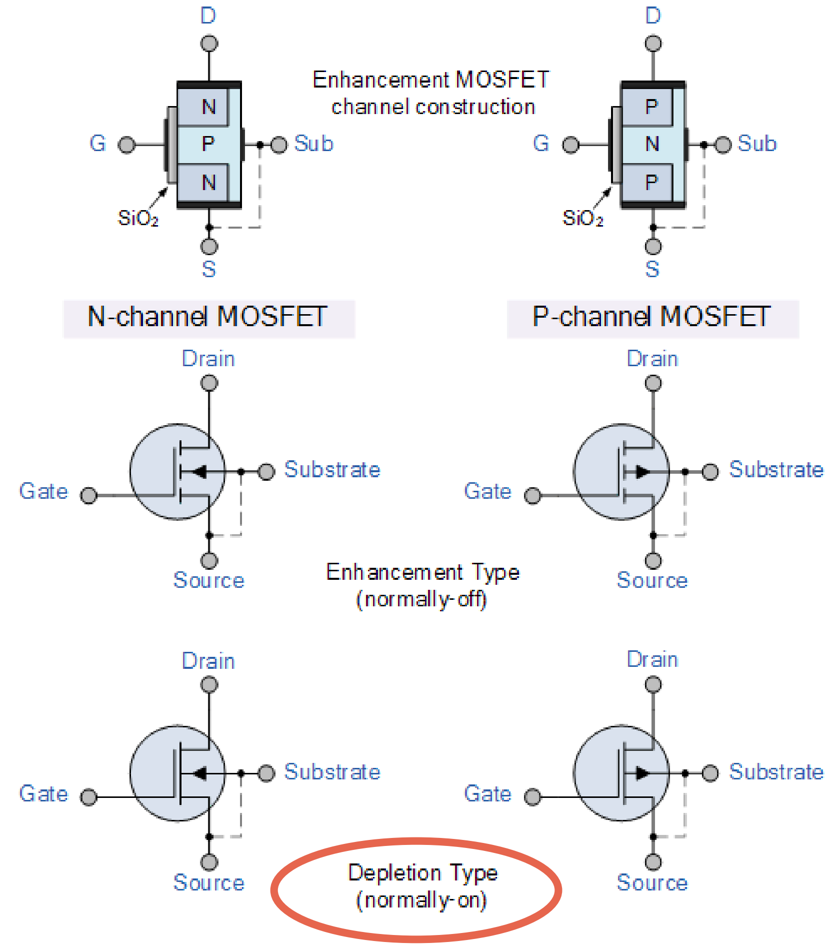
CMOS transistors

- **Voltage** commanded transistors

FET
- JFET
- MOSFET
  - *DEPLETION type*
    - N channel
    - P channel
  - ENHANCEMENT type
    - N channel
    - P channel

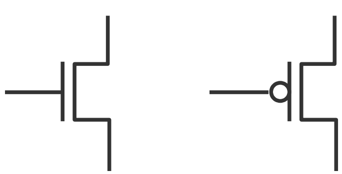
Vgs = 0v → Transistor ON

Vgs > *Vgs threshold* → Transistor ON



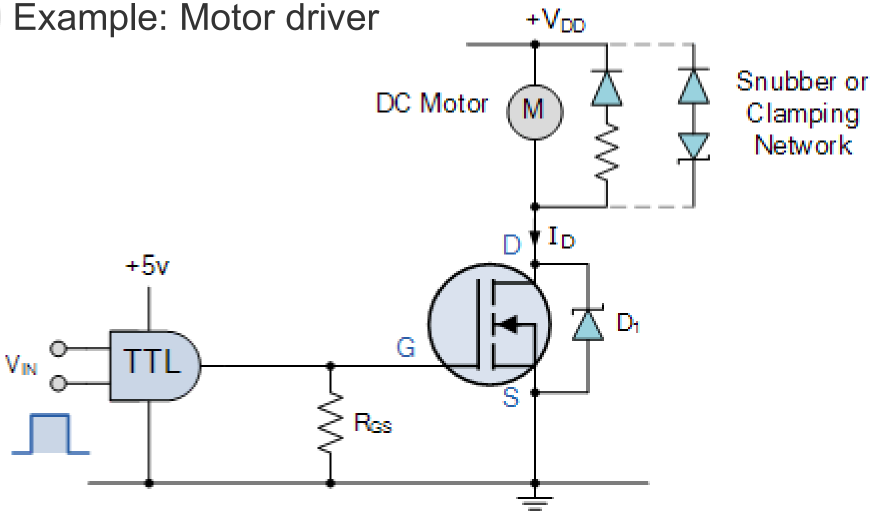
http://www.electronics-tutorials.ws/transistor/tran_1.html

- Common symbols



Nmos Pmos

- Example: Motor driver

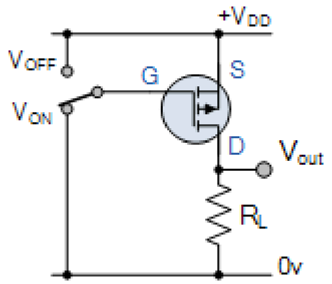


http://www.electronics-tutorials.ws/transistor/tran_7.html

Considering a NMOS enhanced ***2N7002*** *transistor: (see datasheet link below)*

- *Which applied Gate voltage will turn ON the actuator ?*
- *What RGS resistor is useful for ?*

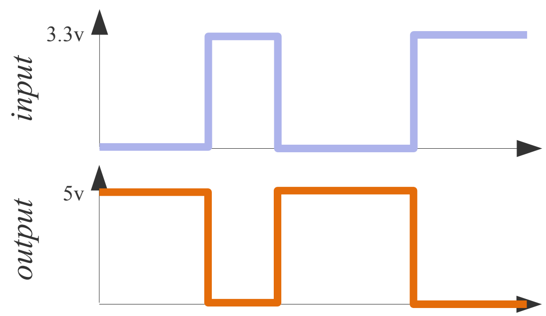
- Example: P-MOS driver



- 2N7002 Datasheets

http://m2siame.univ-tlse3.fr/teaching/francois/NMOS-transistor_2N7002.pdf

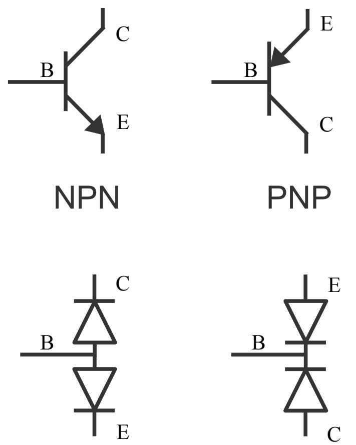
- Your first electronic schematic: *Draw a signal inverter*



… and what about a NAND gate ?

Bipolar transistors

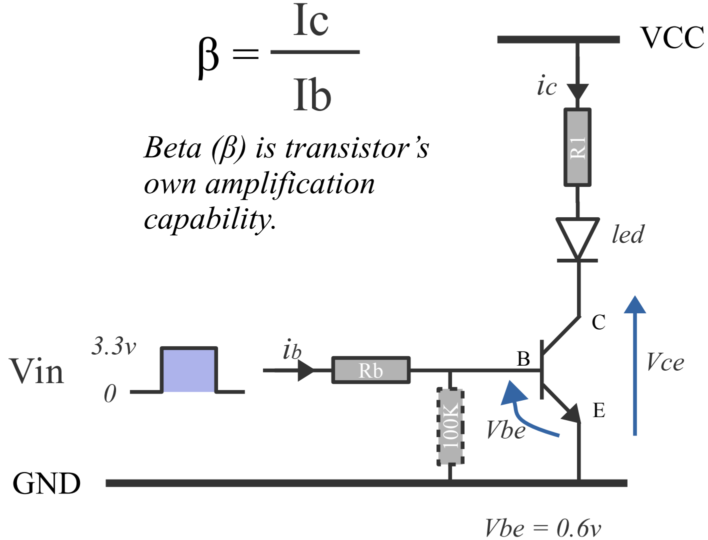
- **Current** commanded transistors



equivalent schematic

$$\beta = \frac{Ic}{Ib}$$

Beta (β) is transistor's own amplification capability.



Q *Considering a NPN BC547 transistor with Vf led = 1.2v, ic = 20mA*

- *Compute R1 and Rb values*

- You own schematic to drive a led considering:

Vin=0v → led ON
Vin=VCC → led OFF

[addon] What happens if VCC > Vin ?

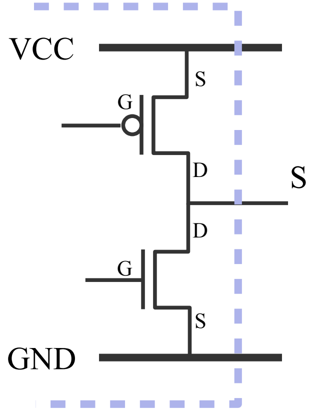
- NPN BC547 Datasheet

http://m2siame.univ-tlse3.fr/teaching/francois/NPN-transistor_BC547-BC550.pdf

http://www.electronics-tutorials.ws/transistor/tran_1.html

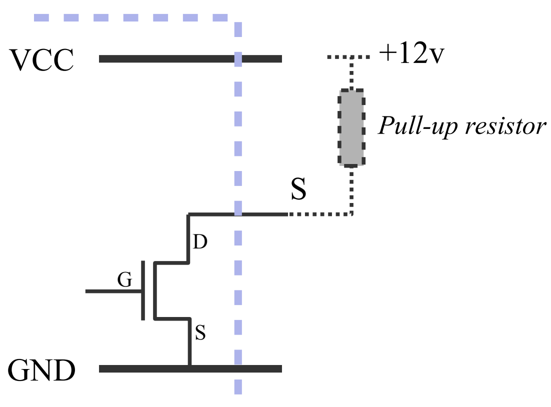
Sink or Source ??

- Totem-pole (*push-pull*) output



Note: same design with bipolar transistors

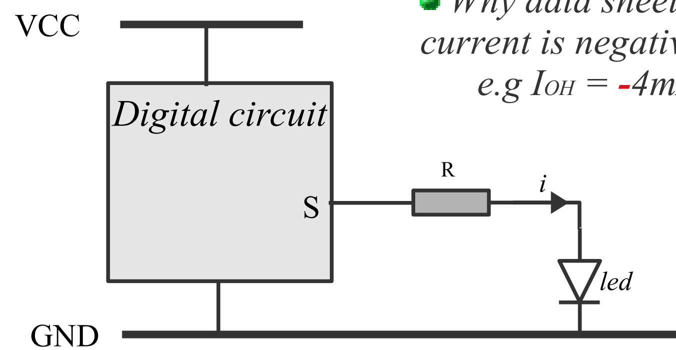
- Open-Drain / Open-Collector output



Note: same design with bipolar transistors

- Source mode:

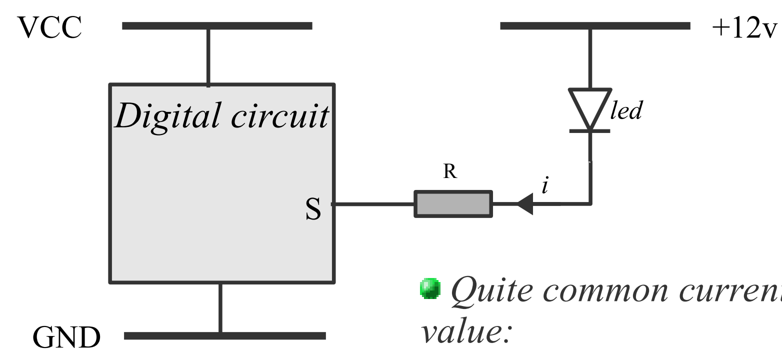
Output provides power to the connected devices



- *Why data sheets tells current is negative ?*
  e.g $I_{OH} = -4mA$?

- Sink mode:

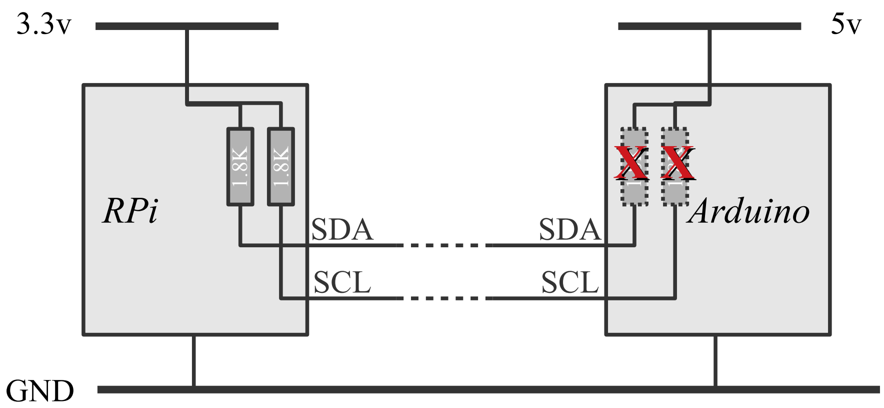
Output absorbs current from connected devices



- *Quite common current value:*
  e.g $I_{OL} = 20mA$

Voltage thresholds

- Application to the I2C bus

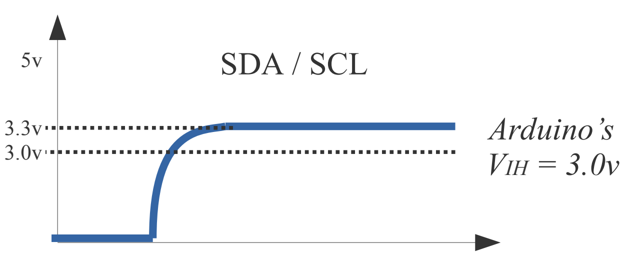


Q

I2C bus master: Raspberry Pi 3.3v powered
I2C bus slave: Arduino 5v powered

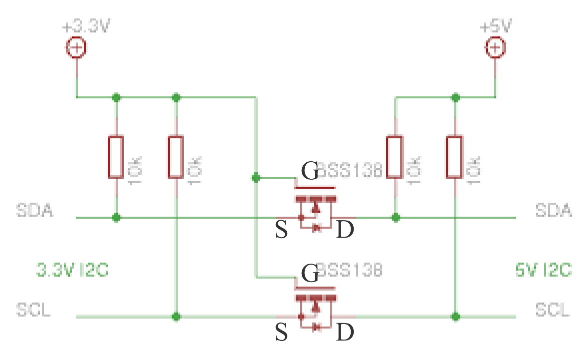
- *Why Arduino and RPi are able to communicate ? What'strouble ahead ??*
- *What happens if you don't remove Arduino's pull-up resistors ?*

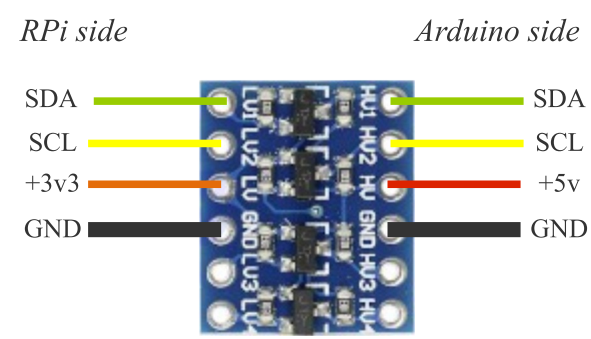
Remember: RPi's GPIO are 3.3v ONLY !!



... hence, only 300mv noise immunity!!

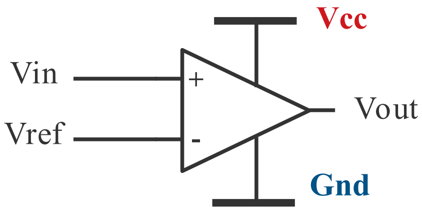
- I2C level shifter





Operational Amplifier

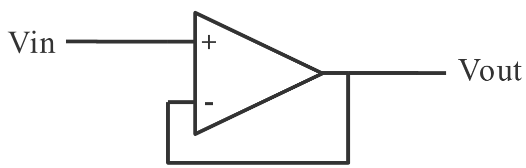
- Comparator



Vout = Ao (V+ - V-)
Note: Ao is open-loop gain

If Vin > Vref then
Vout = VCC
Else
Vout = GND
Endif

- Follower (impedance adaptation level)



Vout = Vin

- Amplifier ...

https://www.electronics-tutorials.ws/category/opamp

Motors

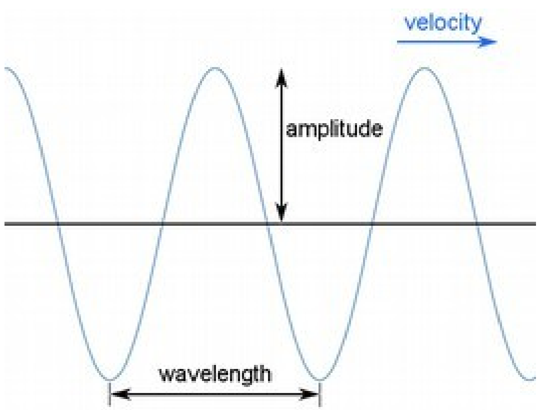
- Servomotor
- Step-motor
- Brushless motor
- H bridge

Wavelengths

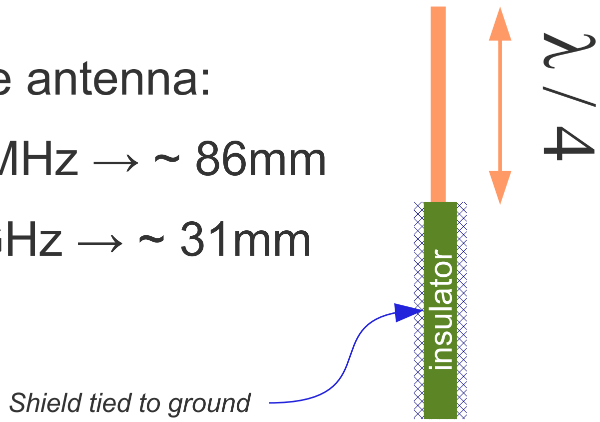
- Antenna lengths calculation

$$\lambda = \frac{C}{f}$$

λ (wavelength) represents the distance over which the wave's shape repeat. (Wikipedia)



λ / 4 wire antenna:
- @868MHz → ~ 86mm
- @2.4GHz → ~ 31mm



- Speeds

C = 300.000 km/s

Photons in an empty space

v = 280.000* km/s

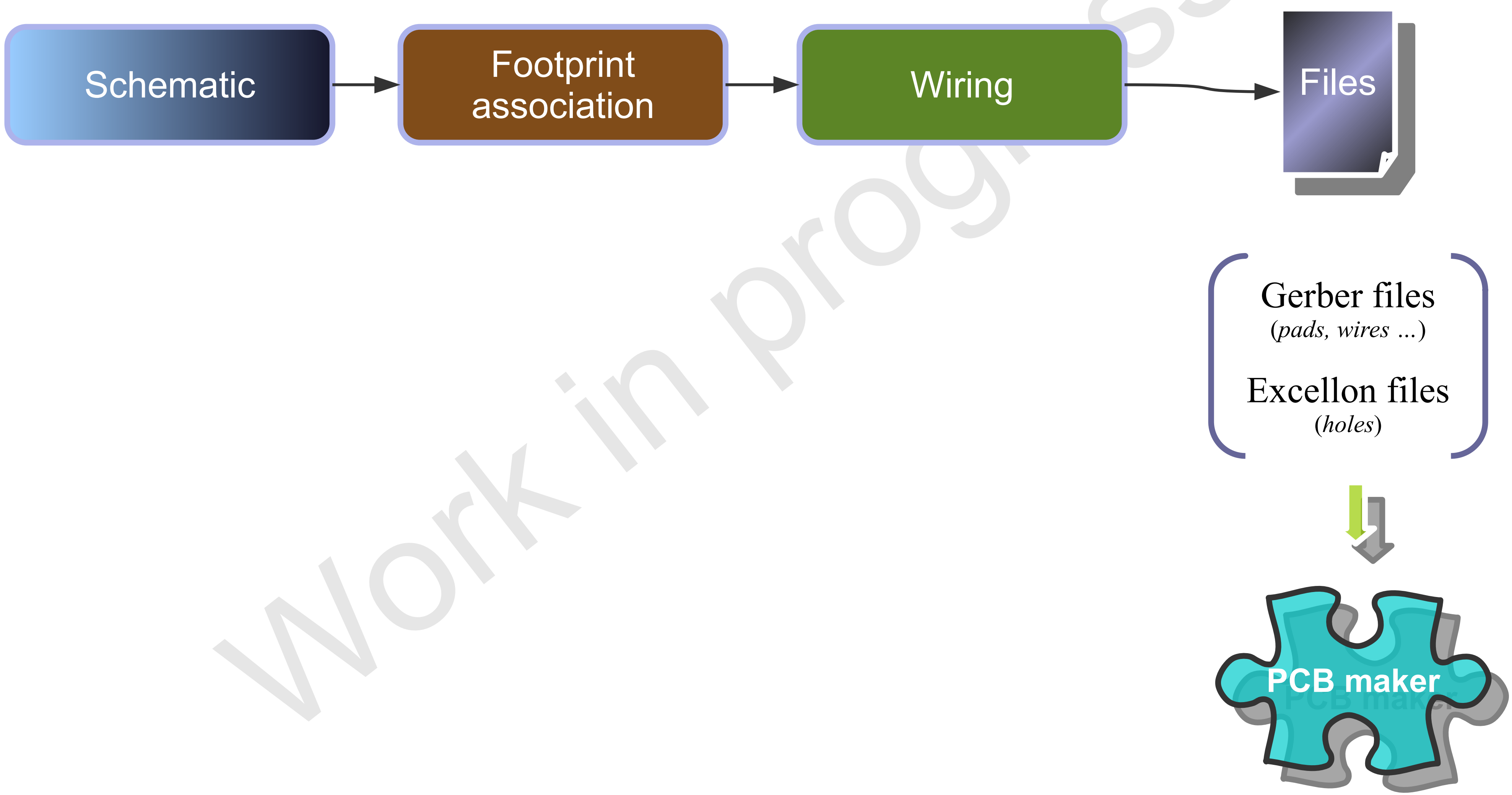
cooper wire

e.g @ 1GHz λ = 28cm

*Electric field propagation in a 12AWG wire. Speed goes from 50 to 99% of C depending on velocity drift.
In a 14AWG → 97% of C

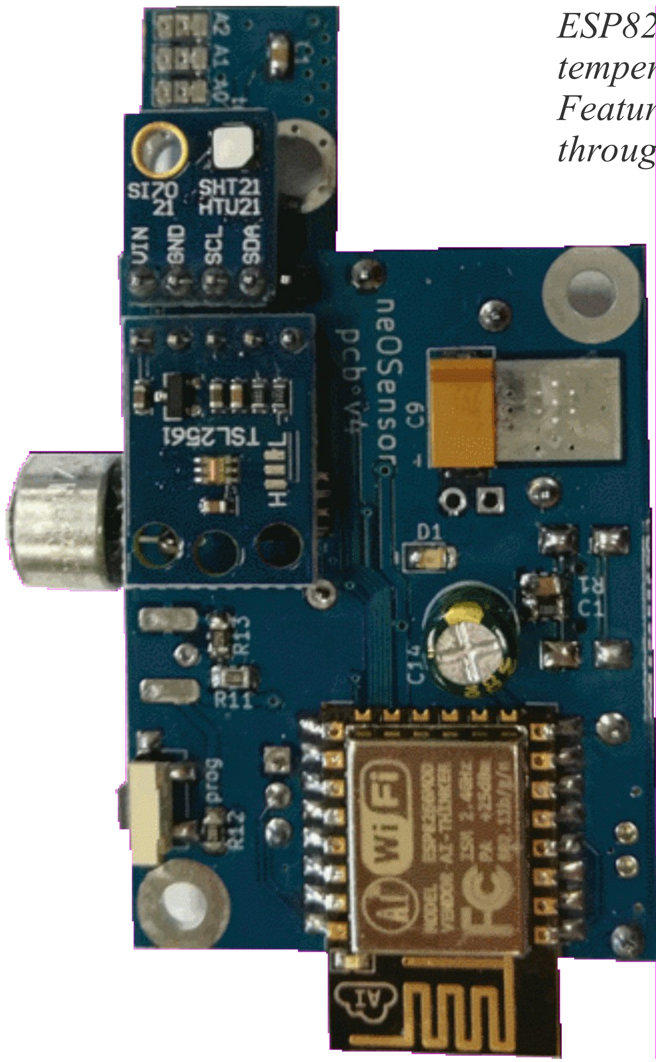
KiCad

- Creating your own **PCB** … *i.e welcome to the master level!*



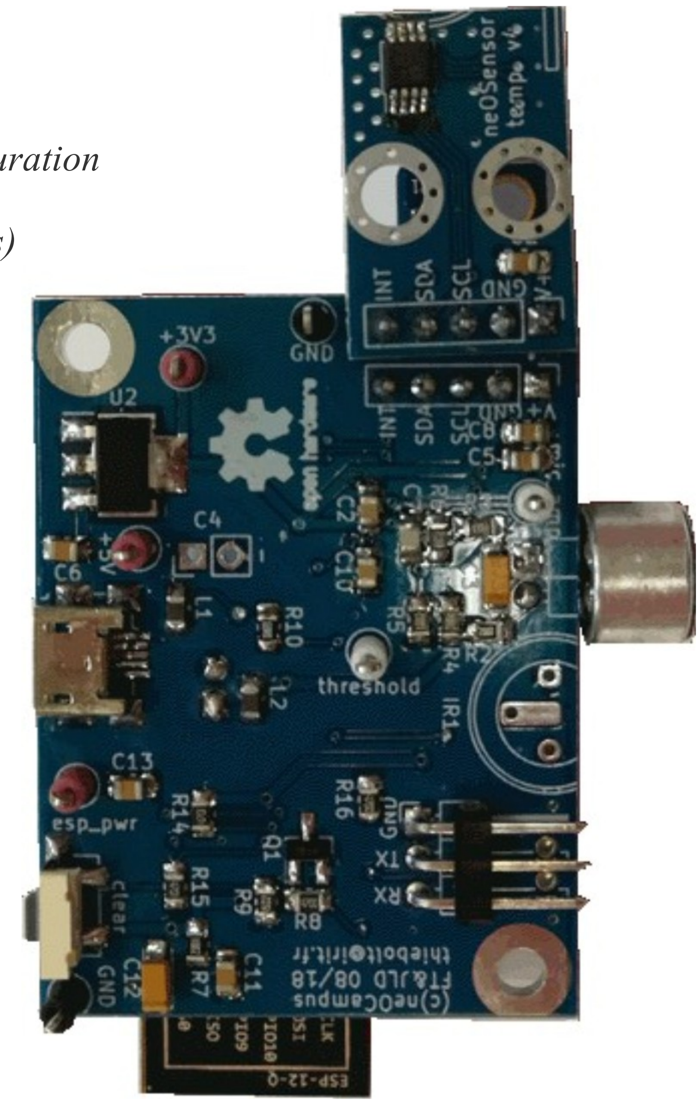
END

- neOSensor: designed by neOCampus with KiCad.
  Files sent to Chinese PCB manufacturer.



ESP8266 based ambient sensor that monitors temperature, humidity, luminosity and noise. Features MQTT, OTA updates and WiFi configuration through embedded HTTP server ~15€.

Deployment @ BU (2016 → nowadays)



Thanks to SCEL@UPS and to the numerous M1 and M2 SIAME students that helped us soldering the boards!