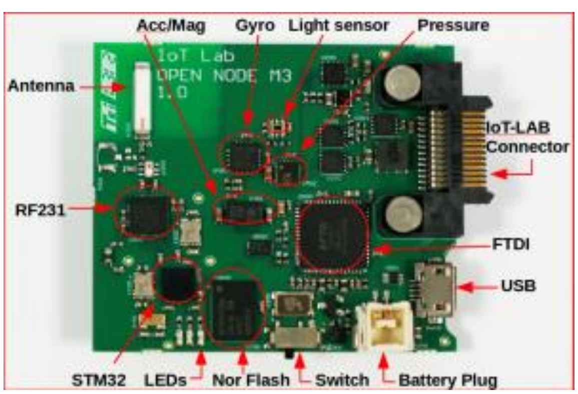
Premises of the 2nd Assignment

IoT-LAB M3

The STM32F401RE MCU used in the first Assignment has been replaced by the ARM Cortex M3 STM32F103REY MCU in this second Assignment.

The M3 open node is based on a STM32 (ARM Cortex M3) micro-controller. Like the WSN node this next generation contains a set of sensors and a radio interface. Main evolutions are a more powerful 32-bits processing, a new ATMEL radio interface in 2.4 Hz and more sensors.

MCU	**ARM Cortex M3, 32-bits, 72 Mhz, 64kB RAM – STM32F103REY**
sensors	• Ambient sensor light – ISL29020 • Atmospheric pressure and temperature – LPS331AP • Tri-axis accelerometer/magnetometer – L3G4200D • Tri-axis gyrometer – LSM303DLHC
radio communication	802.15.4 PHY standard, 2.4 Ghz – AT86RF231
power	3,7V LiPo battery, 650 mAh – GMB 063040



In order for an embedded board to communicate in IPv6 with a host on the Internet, it needs an IPv6 **global** unicast address. To do this, another embedded board play the role of Border Router (BR) and must be added in the network. It will be responsible for propagating an IPv6 global prefix and assign an address to the device.

We speak here of BR, because it’s on the border between a radio network (e.g. 802.15.4) and a classic Ethernet network. Technically, the IPv6 traffic between the SSH frontend and the BR radio interface is routed via a virtual network interface (i.e. TUN or TAP interface) and the IPv6 traffic is encapsulated on the BR’s serial link.

Selecting an already used prefix may bring to an *“overlaps with routes”* error while creating the IPv6 over serial interface - using ETHOS (Ethernet Over Serial). To see currently used IPv6 prefixes on a site, use this command from its SSH frontend:

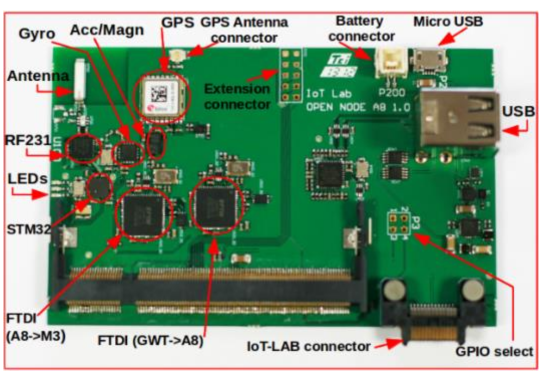
```
<login>@grenoble:~$ ip -6 route
```

Each site has a /64 IPv6 prefix, in which each embedded Linux node will get its static IPv6 address. Its IPV6 configuration can be known with the following environment variables:

```
root@node-a8-1:~# printenv
...
GATEWAY6_ADDR=2001:660:4701:f080:ff::
INET6_ADDR=2001:660:4701:f080::1/64
```

IoT-LAB A8-M3

The IoT-LAB A8-M3 board is based on a TI SITARA AM3505 which is a high-performance ARM Cortex-A8 microprocessor.

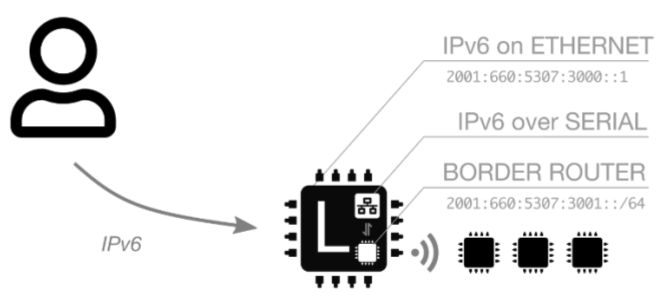


The IoT-LAB A8-M3 embed a clone of the IoT-LAB M3 object (M3 co-microcontroller). As a result, it features the same sensors/actuators, as well as the same radio chip, enabling it to communicate wirelessly with other 802.15.4 objects within radio range. The M3 is linked to the A8 via the I2C data bus and the GPIO inputs/outputs. This M3 co-microcontroller can be programmed from the A8 through a USB/FTDI component.

Mixing two types of boards

It is possible to test common IoT scenarios that implies communications between sensors and an application gateway, the later needing to have a more powerful environment. Boards based on a microcontroller play the role of sensors and a board running embedded Linux, equipped with a co-microcontroller managing a radio interface, plays the role of the gateway.

In that case, it’s therefore quite possible to build an IPv6 network with the co-microcontroller acting as a Border Router and communicating with other nodes by radio (e.g. 802.15.4). Unlike the setup with a microcontroller on the SSH frontend, the creation of the virtual network interface and IPV6 traffic encapsulation is done directly on the embedded Linux node using the co-microcontroller serial’s link.



Ethernet

The IoT-LAB A8-M3 board features an Ethernet interface, enabling it to connect to a LAN and to communicate with the internet. The boards are only accessible by SSH in IPV4 via the SSH frontend and in IPV6 from the Internet.