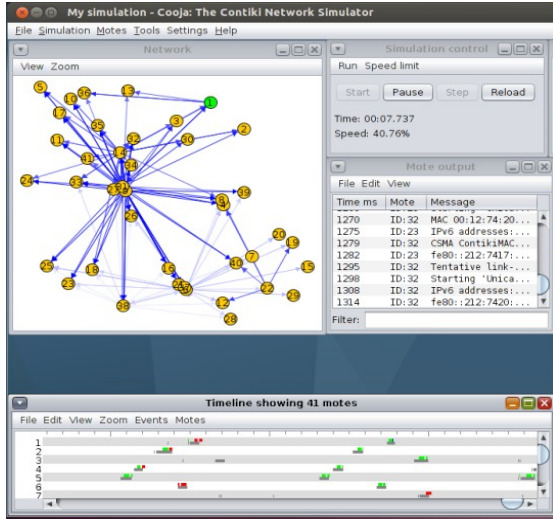


---

# Contiki

The Open Source OS for the Internet of Things



COOJA Simulator

---

**Contiki is an open source operating system for the Internet of Things**

---

Used in a wide variety of systems such as city street lights, networked electrical power meters, remote house and industrial monitoring etc. Contiki allows tiny, battery-operated low-power systems communicate with the Internet.



COCONODE

Tutor : Didier Donsez

Client : Bernard Tourancheau - Malissa Yucinic

## Students :

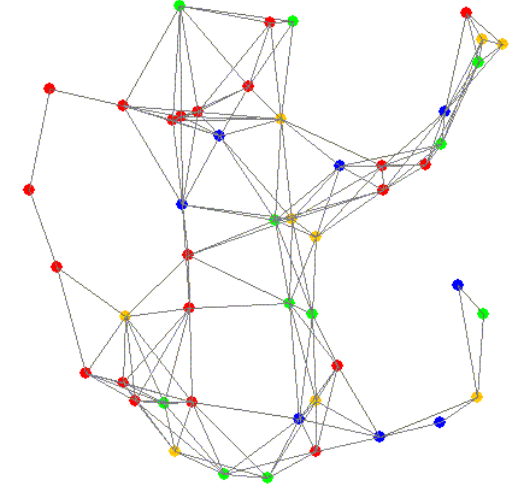
Florian Lévêque

Minh Quan Ho

Noé-Jean Caramelli



RICM 5 – Réseaux Informatiques & Communication Multimédia



---

# COCONODE

**COntiki COoja NODE**

**Wireless Sensor Network (WSN)**

**Simulator**

---



## Why choose Contiki?

- Internet Standards
- Rapid Development
- Low-power wireless devices supported
- Active Community
- Open Source Software
- The Cooja Network Simulator

Cooja, the Contiki integrated network simulator, makes tremendously easy by providing a simulation environment to run applications in large-scale networks and in extreme detail on fully emulated hardware devices.

-- [www.contiki-os.org/](http://www.contiki-os.org/)

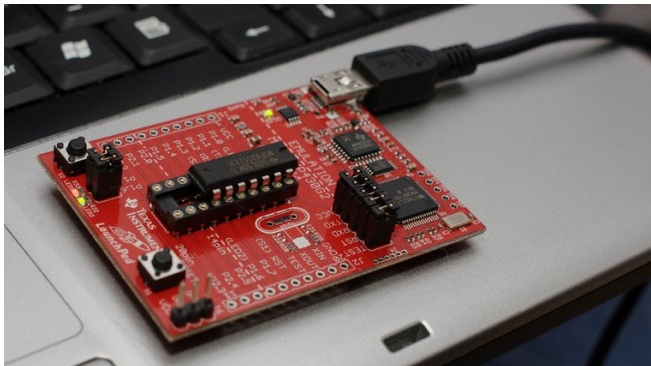
- IETF protocols supported (6lowpan, RPL, CoAP)

- Hardware Manufacturers

Redwire Econotags, Zolertia,

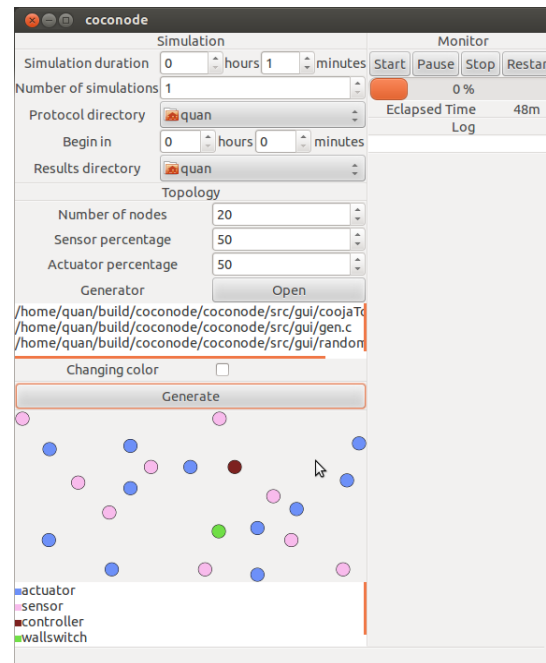
ST Microelectronics,

Texas Instruments, Seed eye



TI MSP 430

## COCONODE



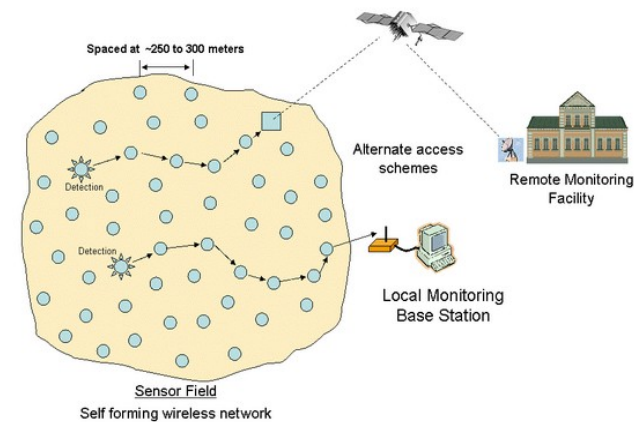
COCONODE GUI

## COCONODE Features

- User-defined Topology Generator
- User-defined Routing Protocol
- JSON configuration import/export
- COOJA-based kernel
- Topology Visualization
- Simulation Loop
- Simulation Scheduling
- Result GNUplot Statistic Reporting

## Specifications

- Modular architecture
- GTK+ based interface, nature and friendly
- JSON-Glib configuration file parser
- COOJA-based C program with Java Native Interface (JNI)
- Dynamic library compilation
- User-defined topology generator loader
- User-defined routing protocol loader



## Keywords

Contiki OS, Cooja, JAVA, JNI, C, JSON-Glib, GTK+, GTK\_Thread