S4 project

The HomeSIP Project: home automation with SIP

Salim ELLOUZE
Damien LIGOT
Julien MARCHET
Hanitra RATRIMO

Ω-SIP

Supervisor: Patrice KADIONIK

ENSEIRB
ECOLE NATIONALE SUPERIEURE
ÉLECTRONIQUE, INFORMATIQUE & RADIOCOMMUNICATIONS
BORDEAUX

The HomeSIP project

- HomeSIP project overview
- SIP protocol
- Development environment
- Communication with SIP
- Graphical User Interface (GUI)
- Conclusion

The HomeSIP project

- HomeSIP project overview
- SIP protocol
- Development environment
- Communication with SIP
- Graphical User Interface (GUI)
- Conclusion

Home automation with SIP (1/4)

SIP : Session Initiation Protocol

•Collaboration between IMS, LABRI and ENSEIRB:

•**Global Objective:** Set up a sensor network for Home Automation that will be monitored remotely by smart devices using SIP as a container for collecting data

S4 project specific objective: develop embedded software with a GUI on a Nokia n770 PDA in order to communicate with a temperature sensor using the SIP protocol.

Home automation with SIP (2/4)

SIP : Session Initiation Protocol

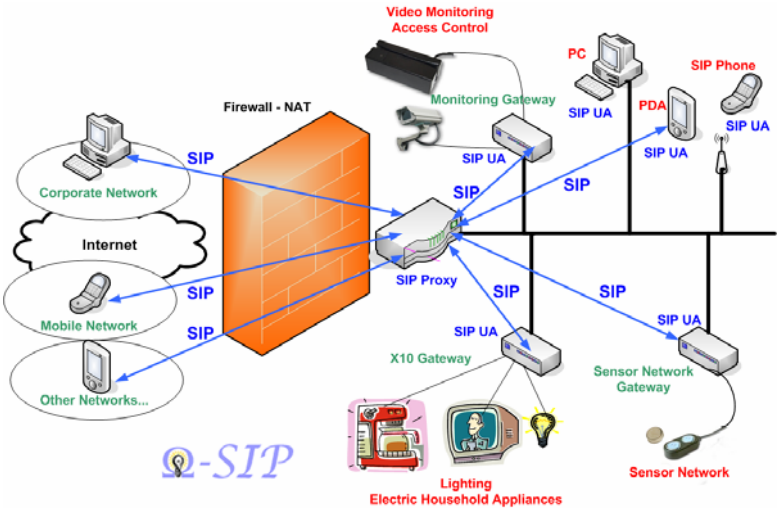
- **The HomeSIP project has 3 main goals:**

1.Setting up an hardware platform made of sensors and actuators connected to Linux embedded systems with network capabilities

2.Developing specific software on Linux embedded systems implementing the SIP protocol to communicate with the sensors

3.Creation of a new DSL language (Domain Specific Language) made to develop new services on the HomeSIP platform.

Home automation with SIP (3/4)



The HomeSIP Hardware platform

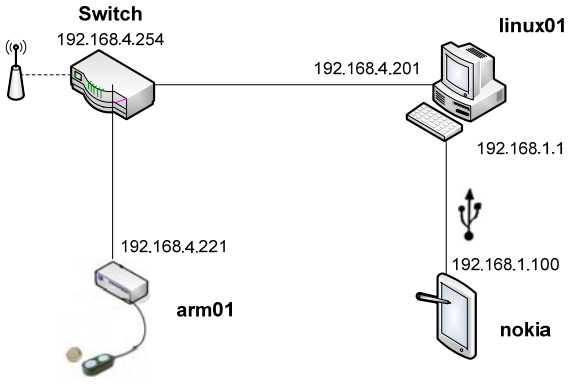
Home automation with SIP (4/4)

The sensor hardware gateway which is operational today is based on commercial hardware:

- An ARM9 SBC board running embedded Linux with I/O connections (USB, serial line) and an Ethernet interface to act as a SIP gateway.

- An i-button DS1920 temperature sensor from Dallas Semiconductors connected via serial line connection.

Hardware architecture



Objective: a user-friendly graphical interface on PDA Nokia n770 to check the temperature

Project overview

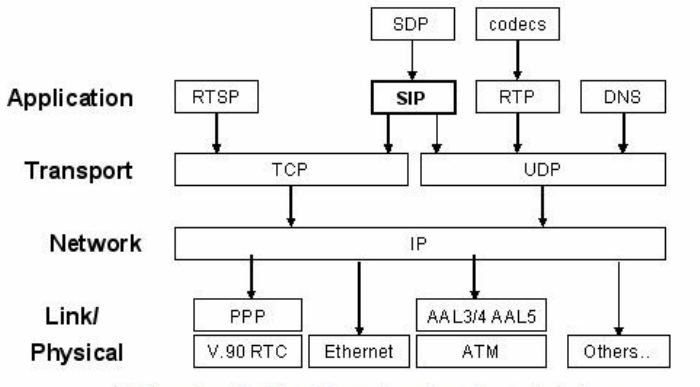
- 3 main steps:
  - 1. Set up the cross-compilation environment for nokia n770 (Scrachbox, Maemo, osip/eXosip libraries)
  - 2. Develop client/server applications using the osip stack (command line applications)
  - 3. Develop the GUI using Hildon environment and GTK+ graphical libraries

The HomeSIP project

- HomeSIP project overview
- SIP protocol
- Development environment
- Communication with SIP
- Graphical User Interface (GUI)
- Conclusion

SIP protocol (1/3)

• The Session Initiation Protocol (SIP) is an application-layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants.



SIP protocol in the Internet protocol constellation

SIP protocol (2/3)

- Mostly used for Voice over IP (VoIP) or instant messaging (GAIM, MSN)
- Several qualities for Home automation:
  - it supports different communication mechanisms for devices
  - it works with different in-home networking technologies : PLC, X.10, CAN bus
  - it provides a flexible payload capability based on MIME types
  - it can reuse existing SIP infrastructures for new services
- Best choice compared to other protocols such as SNMP or HTTP

SIP protocol (3/3)

The SIP stack

- oSIP stack and its extension API eXosip
- **Osip capabilities:** 
  - low level SIP signaling
  - it supports many transport protocols such as TCP, UDP and TLS
- **eXosip capabilities:** 
  - high layer for SIP signaling for softphone or endpoint server.
- written in C language, very portable and very low footprint

The HomeSIP project

- HomeSIP project overview
- SIP protocol
- Development environment
- Communication with SIP
- Graphical User Interface (GUI)
- Conclusion

Development environment

Development platform

Cross-compilation using arm-linux-gcc compiler

Cross-compilation using Scratchbox

- osip and eXosip had to be compiled for both CPU architecture before compiling applications.
- Therefore, the libraries path had to be specified in the compilation command:

```
arm-linux-gcc -O2 -g -I/home/guest/arm/include -
  L/home/guest/arm/lib -DENABLE_TRACE $1.c  -o $1 -leXosip2 -
  losip2 -losipparser2 -lpthread
```


Development environment

SIP

Protocol

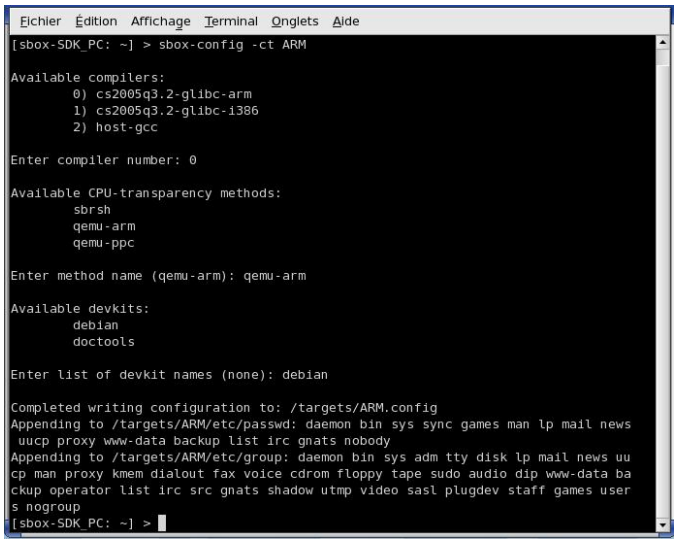
- Generated libraries and server application files were sent using minicom, a terminal emulation program, via serial line.
- We used nfs protocol to mount the computer file system on the ARM.

- Generated libraries and client application files were sent using ssh protocol.

Scratchbox

- Scratchbox was developed by Movial and sponsored by Nokia.
- Scratchbox is a cross-compilation toolkit designed to make embedded Linux applications easier.
- It provides set of tools to integrate and cross-compile an entire Linux distribution.
- It supports many architectures like ARM and x86.
- By installing Maemo rootstrap, it is possible to compile Maemo applications on both ARM and x86 architectures.

Scratchbox



Creation of the ARMEL compilation environment on Scratchbox.

Xephyr

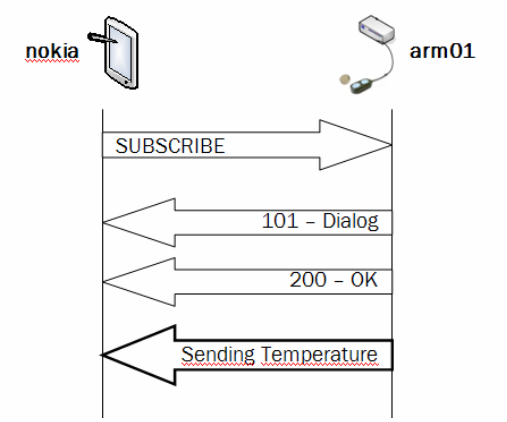
- Xephyr is a drive-based X server which targets a graphical window on a host X server as its framebuffer.
- It was required to see and test our application on the computer.

The HomeSIP project

- HomeSIP project overview
- SIP protocol
- Development environment
- Communication with SIP
- Graphical User Interface (GUI)
- Conclusion

Communication with SIP (1/4)

Theory



The client (nokia) wants the server to send the temperature. After the establishment of the connection , the server sends temperature regularly to client.

Communication with SIP (2/4)

Client (Nokia)	Server (Arm)
	-Initialization -Waiting for a request
-Initialization -Sending of a subscribe request Target SIP address: `sip:root@target` Host SIP address: `sip:root@host` Communication duration: `Expiration_time`	
	-Reception of the message -Extraction of the client SIP address and expiration time -Getting of the temperature from the i-button
-Reception of the temperature and display	-Sending of a message with the temperature in the body for the communication duration
-End of the program	-Pending for another request

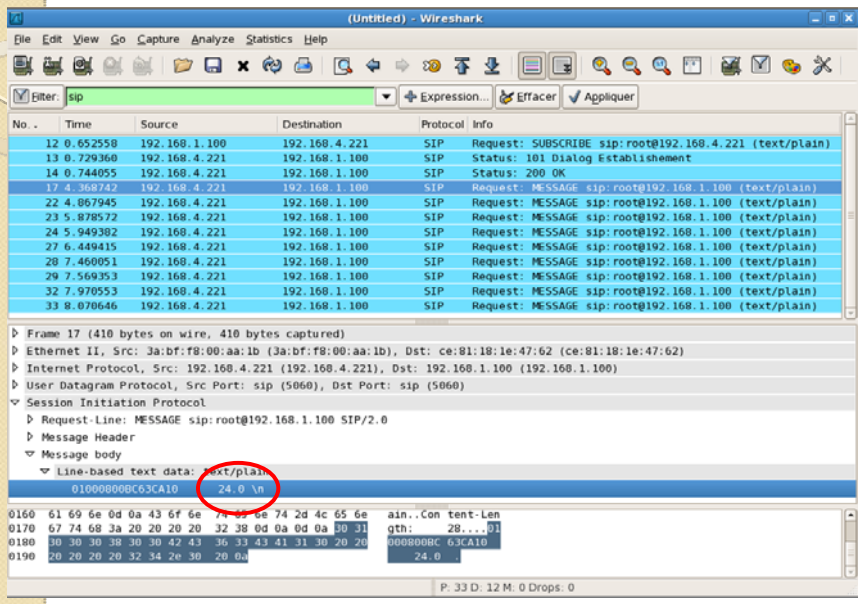
Communication with SIP (3/4)

Execution :

Nokia (client)	Arm01 (server)
~#./test_im sip:root@192.168.1.100	# ./send_notify_im
sip:root@192.168.4.221 30	
Annonce 1xx	200 OK Response Building OK
Annonce 200 OK	200 OK Response Sending OK
01000800BC63CA10 28.0	Temperature : 28.0
01000800BC63CA10 28.1	Temperature : 28.1
01000800BC63CA10 28.1	Temperature : 28.1
01000800BC63CA10 28.2	Temperature : 28.2
01000800BC63CA10 28.3	Temperature : 28.3
01000800BC63CA10 28.4	Temperature : 28.4
01000800BC63CA10 28.4	Temperature : 28.4
01000800BC63CA10 28.4	Temperature : 28.4
01000800BC63CA10 28.6	Temperature : 28.6
01000800BC63CA10 28.6	Temperature : 28.6
01000800BC63CA10 28.7	Temperature : 28.7
01000800BC63CA10 28.8	Temperature : 28.8

Each temperature is correctly sent, received and displayed.

Communication with SIP (4/4)



The HomeSIP project

- HomeSIP project overview
- SIP protocol
- Development environment
- Communication with SIP
- Graphical User Interface (GUI)
- Conclusion

Graphical User Interface

Objectives :

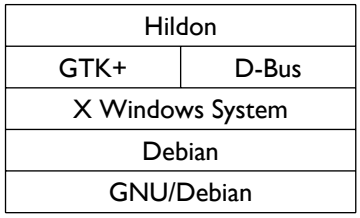
- Adapted to Nokia n770 PDA
- User-friendly and simple
- Improvable
- Easily updatable

Maemo SDK (1/2)

- Open-source platform for Linux-based handhelds
- Eases software development for embedded devices
- Practical tool to test and debug applications using a Linux PC

Maemo SDK (2/2)

- Maemo library stack is made up with open source components well known on Linux desktops :

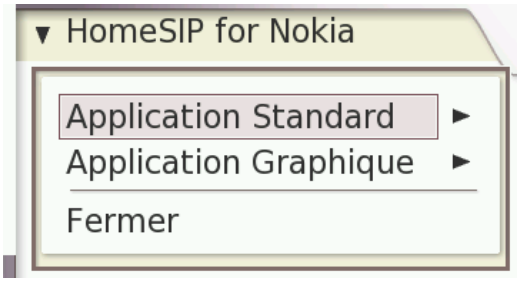


Hildon environment (1/2)

- GUI designed for small mobile devices
- Suggested by Maemo Tutorial 2.2
- Hildon environment is a modified version of GTK+ (GIMP ToolKit) which is more suitable for embedded systems.

Hildon environment (2/2)

- Menu example with Hildon GUI :

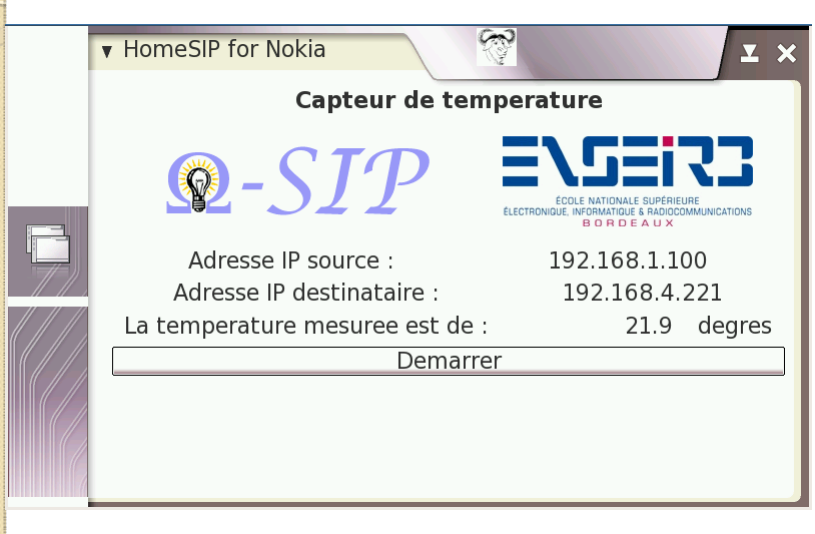


GUI Implementation (1/3)

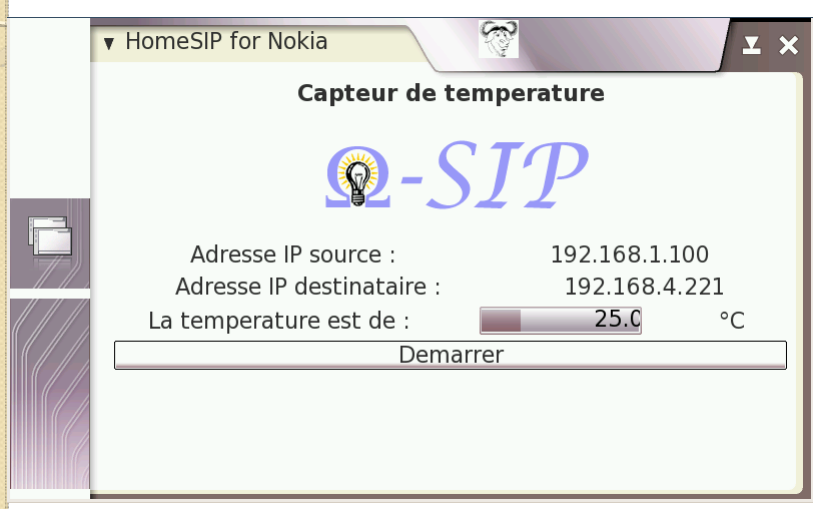
- C programming
- GTK+ widgets and methods mainly used
- Adaptation to Hildon graphical interface
- Tests and debug processes on Maemo platform

GUI Implementation (2/3)

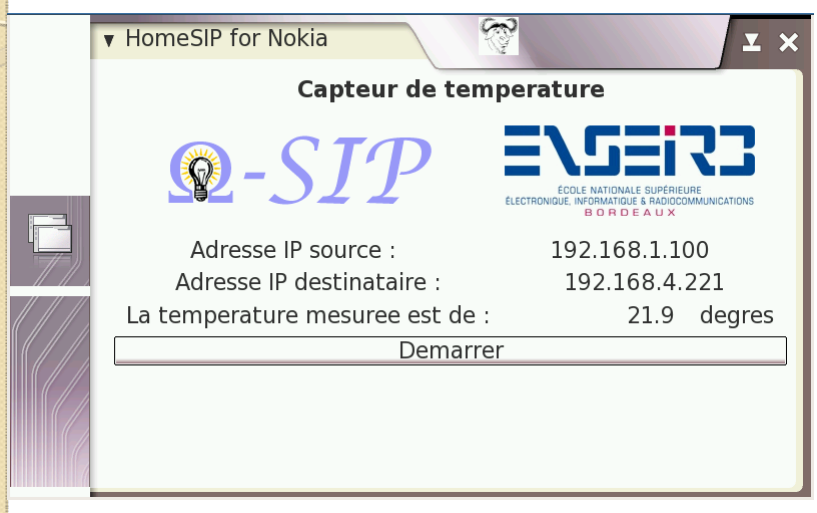
- A few examples of the result:



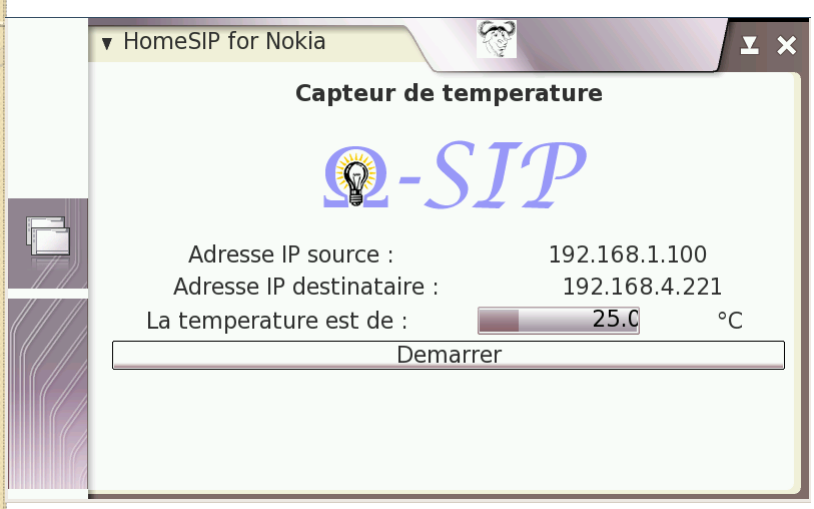
GUI Implementation (3/3)



Graphical User Interface (GUI)



Graphical User Interface (GUI)



The HomeSIP project

- HomeSIP project overview
- SIP protocol
- Development environment
- Communication with SIP
- Graphical User Interface (GUI)
- Conclusion

Conclusion

- Result: working user-friendly graphical interface implemented on a PDA

- Possible improvements:
  - Use of WiFi link rather than USB link

Conclusion

Thank you for your attention