

NewLogic leading WiFi protocol stack developed in SDL-RT with RTDS.



This is the transcript of an interview of Guilhem Roques -820.11 software engineer- done in June 2004.

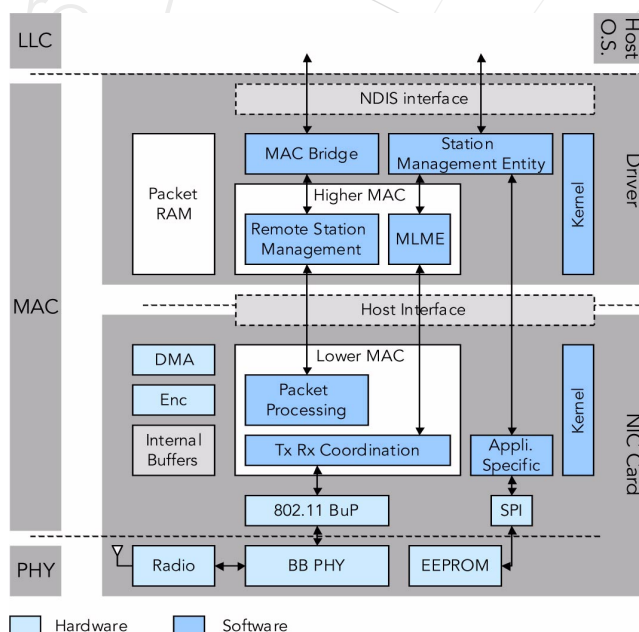
In December 2005, Wipro acquired European System on Chip (SoC) design firm NewLogic.

The company

NewLogic is an Austrian company that has several R&D centers in Europe. Our business is to provide Bluetooth and 802.11 IP (VHDL and C) as well as design services. Our customers are manufacturers looking for a solution to integrate Bluetooth or WiFi solutions within their own product. Generally speaking our product is integrated on a larger chip such as a mobile phone or a PDA.

The project

The 802.11 a/b/g stack has been developed using RTDS in Sophia-Antipolis. The stack includes radio, modem, and software parts. The project started in 2001 and involved a team of 10 people for the software part.



The needs

Following a previous experience the software team was convinced to use a graphical modelling tool for the protocol stack. We started looking for a tool that could:

- generate legible C code,
- help to maintain it through out the project,
- could be integrated with our existing home made scheduler.

The solution

RTDS was selected because it answered all our requirements:

- write finite state machines based on graphical SDL standard,
- generate C code that could be integrated with our kernel,
- for a reasonable price.

The Process

RTDS is used to generate C code and is integrated as is in our product. Other developers that might not use RTDS see the generated C code as some hand-written code. That allows us to keep using our internal utilities such as makefiles and to keep the flexibility of hand-written C code.

Design is very simple because the tool is based on a graphical language, and it pays off when coming back to the design 1 or 2 years later to have a very legible code, easy to follow when debugging. That reduces the amount of problems compared with hand-written C code. When designing it has the flexibility of C in a well defined structure for the state machines based on SDL standard. Time between development and target testing is reduced because of the code generation.

Documentation

RTDS allows to export the diagrams and to integrate them within the documentation. SDL graphical state machines are self documented so time needed for documentation is substantially reduced because state machines do not have to be explained twice: once in the documentation and another time in the code.

PragmaDev

We have very good relationships with PragmaDev that acted like a partner. We had a lot of requests at the start

of the project to be able to optimize and integrate the generate code with our kernel. PragmaDev has been very reactive and careful regarding our needs and problems leading the tool to perfectly match our needs in the end.

Future work

There is no doubt RTDS will be used to develop other products. We strongly recommend RTDS except to our direct competitors (*laughs*). Seriously, it saves time. A lot of developers are suspicious over graphical programming and code generation. It is true it costs some time to get started with the tool but afterwards it saves time every day on test and simulation once you get to trust the generated code.

