

# Debugger for CM-LYNX Software Development Kit



**CUSTOMER:** JSC ICC Milandr



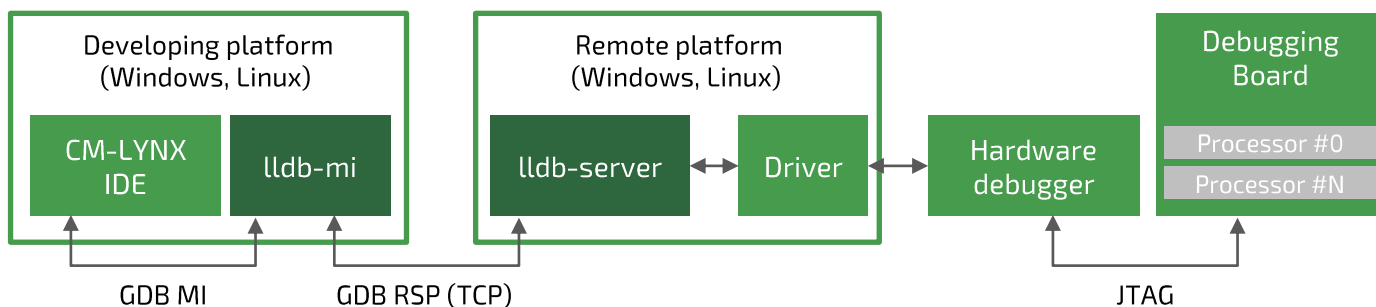
**INDUSTRY:** integrated microcircuits

## PROBLEM STATEMENT:

The debugging application used in the CM-LYNX integrated development environment (IDE) for signal processors did not meet the clients' requirements.

A debugger is one of the major components of a software development kit (SDK).

Milandr used a debugger based on several subsets of GDB debugger source codes, which had a number of limitations. Its further development seemed infeasible and costly due to the "original roots" of the solution, which ignored the results of the open project community.



## SOLUTION:

The project goal was to switch to one of the open debuggers and port it to the LYNX platform.

We considered two options, GDB and LLDB, settling on the latter as it was 100% compatible with the existent compiler based on LLVM for CM-LYNX, which we had already developed for the same framework before.

According to the current CM-LYNX development concept, we set the additional goal of providing a detailed documentation package for this component.

## RESULT:

- LLDB debugger was adapted, its functionality was extended to provide work with CM-LYNX IDE;
- Hardware debugging server based on lldb-server was implemented;
- A documentation set was developed in full compliance with Milandr standards, including software requirement specification, implementation specification, and test specification;
- The debugger was integrated into IDE.

The project contributed to the enhancement of debugger functionality as a part of CM-LYNX product, increased performance stability and significantly cut the cost of the IDE upgrade.

End users highly appreciated the modifications. The updated product with the enhanced debugger reaffirmed Milandr's position as a leading designer and manufacturer of computers running on DSP processors.

*We have been working with LLVM technology for a long time already; however, that was our first experience in porting LLDB debugger. LLDB project was a "young" one and did not seem feasible for commercial use. In recent years, the situation has changed, so our acquired experience and positive results are the reasons why we can recommend LLDB to other our customers – Paul Boiko, Grovetly*

## TOOLS & TECHNOLOGIES:

LLVM, LLDB, JTAG, Eclipse, C++



grovetly.com



sales@grovetly.com