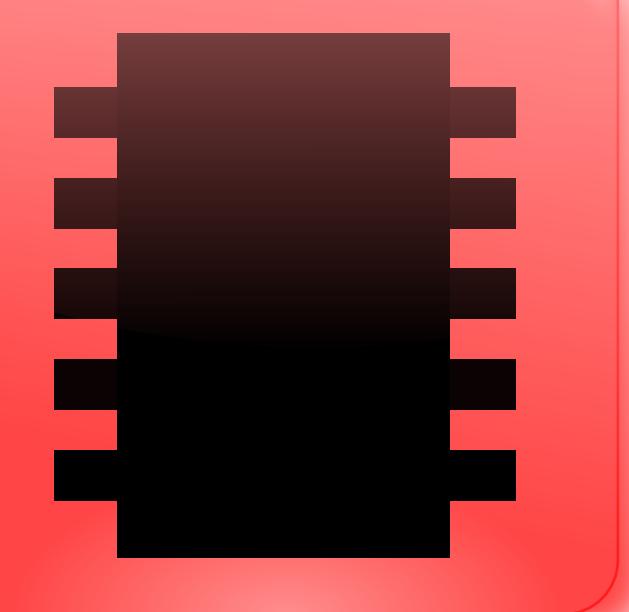


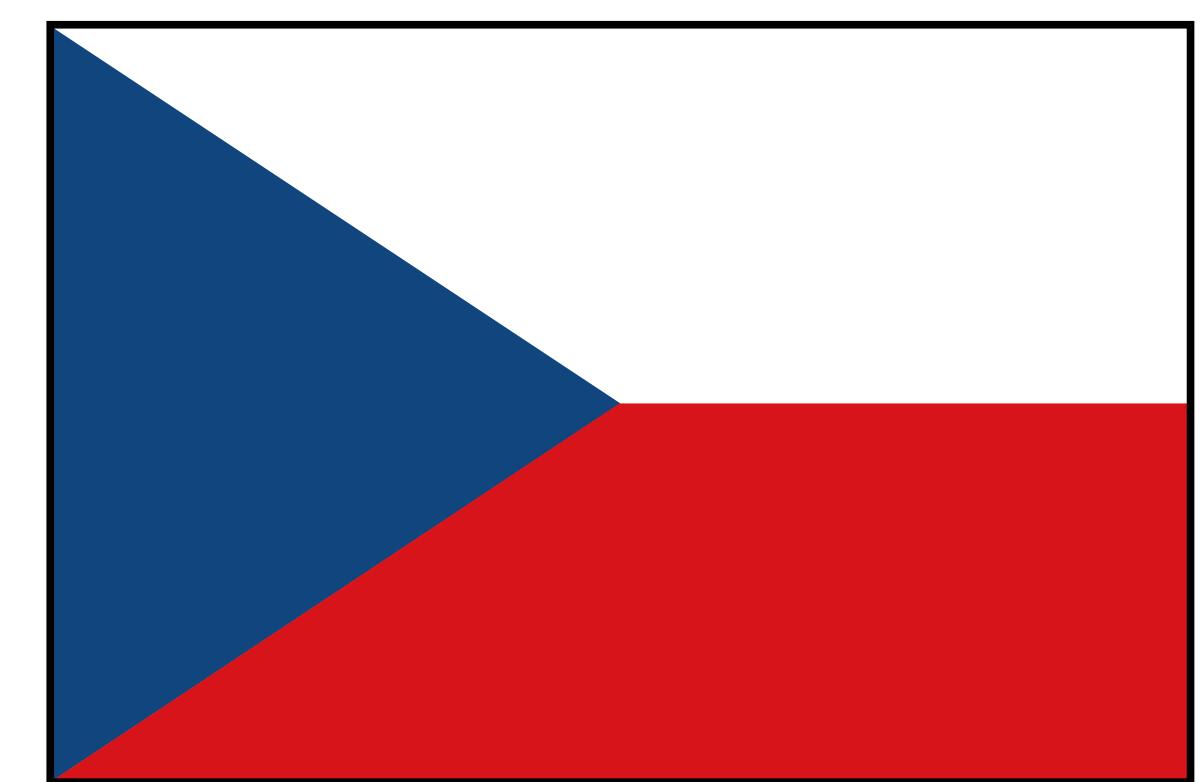
LORRIS TOOLBOX

Microprocessor Application Development Kit



Vojtěch Boček
Czech Republic

Secondary School of Engineering, Brno, Sokolská 1



The goal

While building robots for various student competitions, I've encountered a problem with displaying the data from the robot. It contained huge number of various sensors and there was no program capable of displaying the data produced.

The goal of this project was to create a software capable of **displaying binary data** from various electronic devices **in a clear and well-organized manner**.

About project

Lorris is extensive set of tools designed to help with development, debugging and controlling of various electronic devices, for example robots.

Lorris currently contains four different tools: analyzer, programmer, terminal and proxy server. Each of them is extremely helpful for both programmers and engineers.

Analyzer

Analyzer can parse and display any binary data stream. It uses "widgets": little windows with specific piece of data - number, angle, bar, color, graph and more. Analyzer can also send data back to the device, which can be used to for example control robots.

The Analyzer's environment is fully scriptable, which allows users to process and display nearly any type of input data and create any kind of controls for their application.

How does analyzer work?

Microchip



Binary data stream

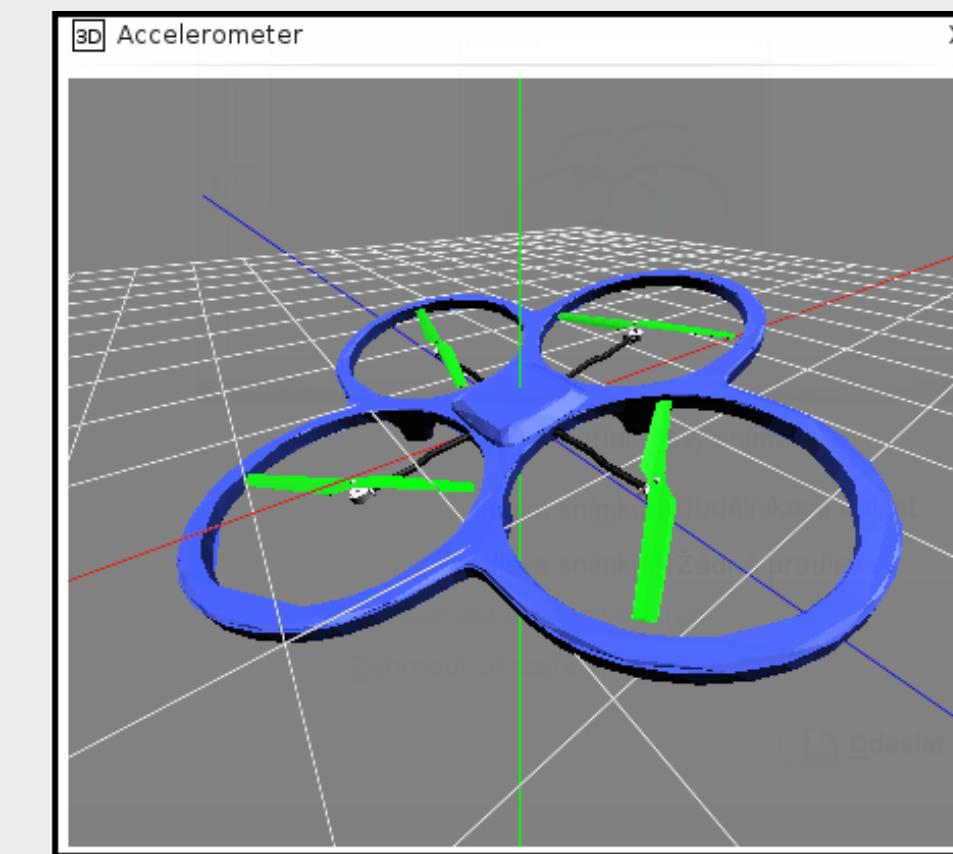
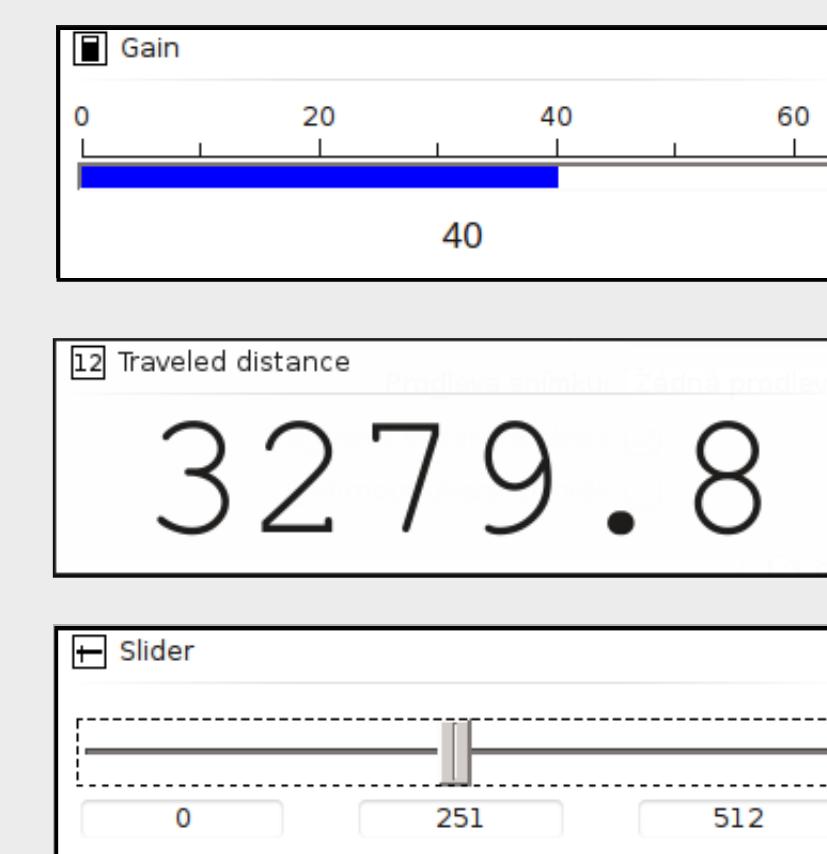
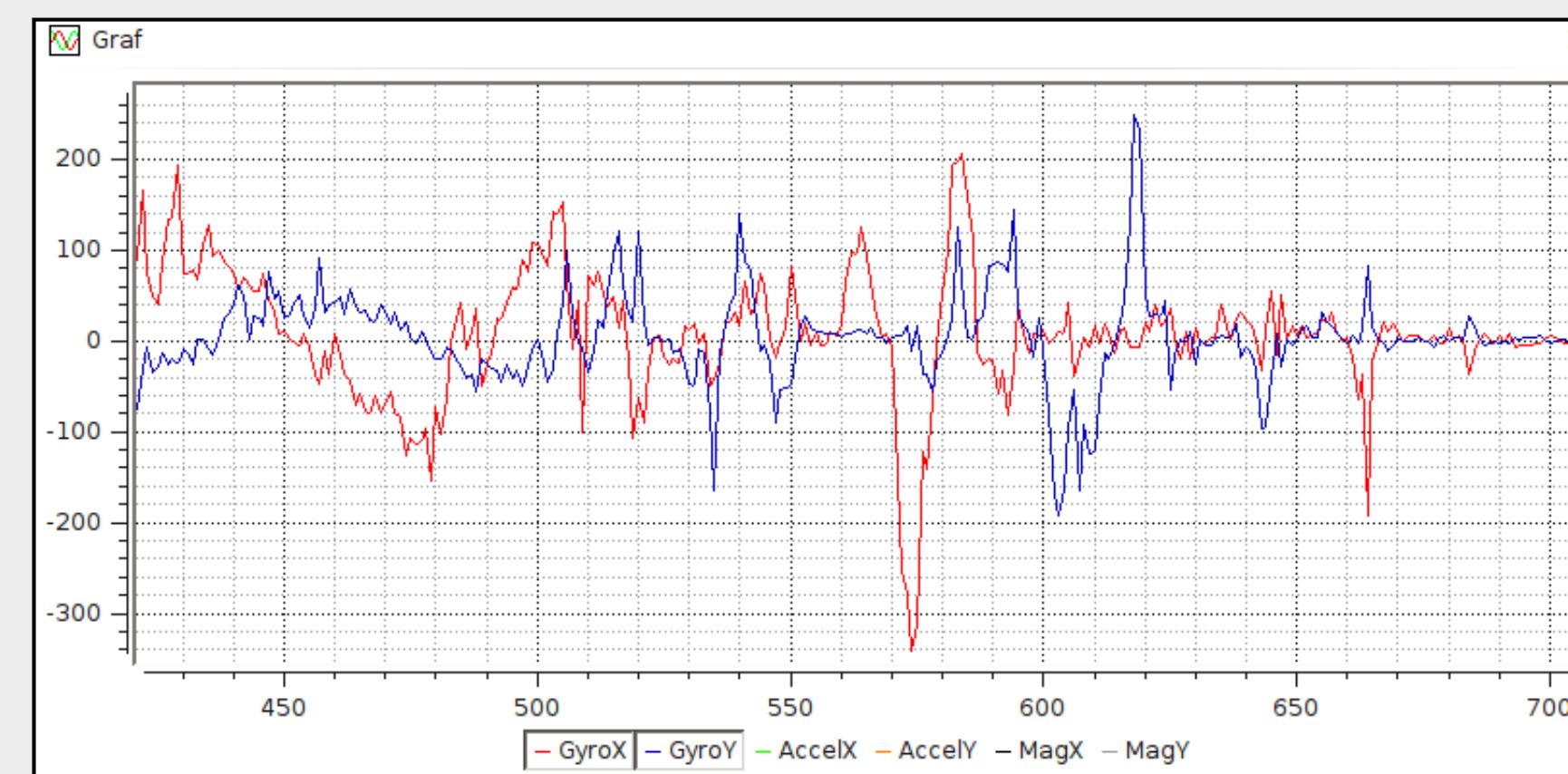
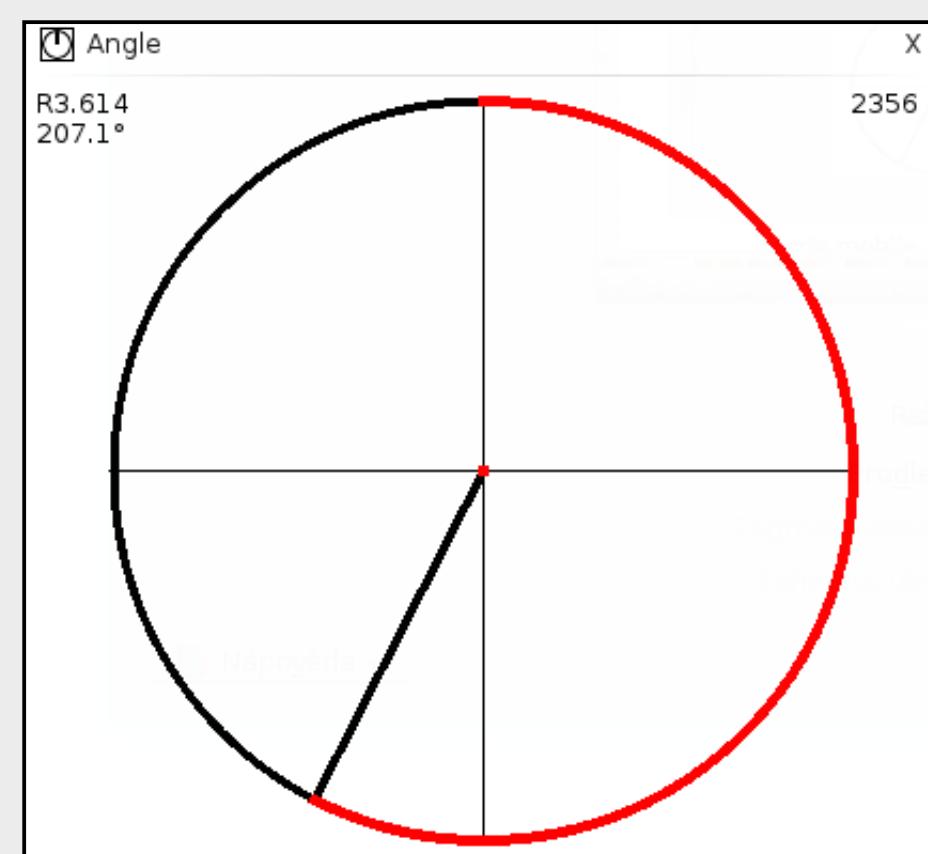
```
00000010 6E 70 C8 58 F9 40 E7 30 C5 A0 00 00 06 26 73 6E
00000020 70 C8 56 00 14 FF C8 00 11 00 00 04 5B 73 6E 70 C8
00000030 C8 5A 00 15 C5 01 07 00 00 03 55 73 6E 70 C8
00000040 5C 00 1E 00 15 FD DA 00 00 04 81 73 6E 70 C8 5E
00000050 F0 B2 F7 88 15 00 00 06 A4 73 6E 70 C8 60 00
00000060 BA 06 D5 09 1F 00 00 04 36 73 6E 70 C8 F9 50 E7
00000070 FC 41 B9 00 00 06 17 73 6E 70 C8 F9 50 E7
00000080 00 C5 C0 00 00 06 26 73 6E 70 C8 56 00 04 FF C2
00000090 00 00 04 47 73 6E 70 C8 5A 00 17 00 CA 01 03
```

Processing

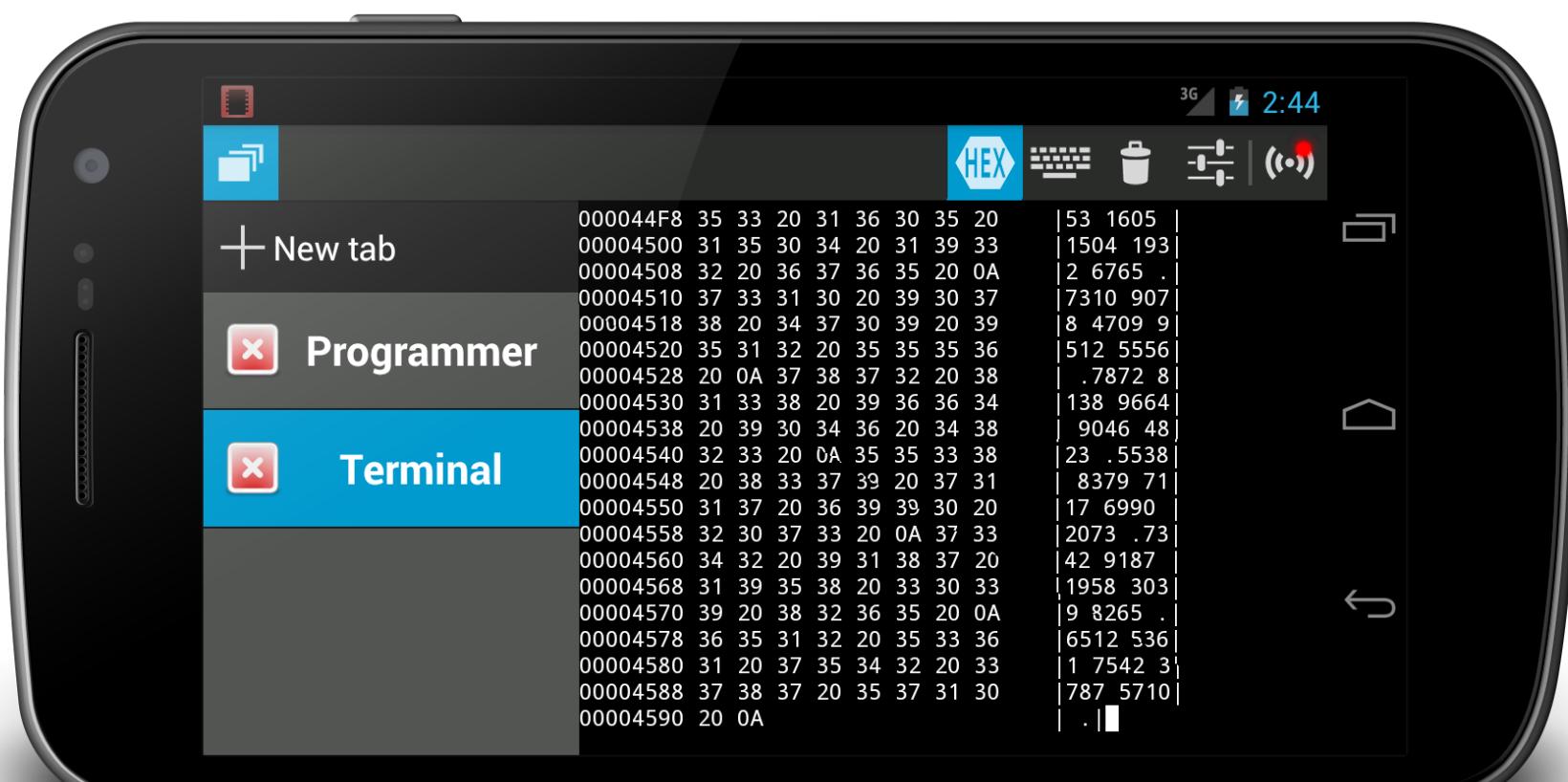


Lorris

Displaying and control



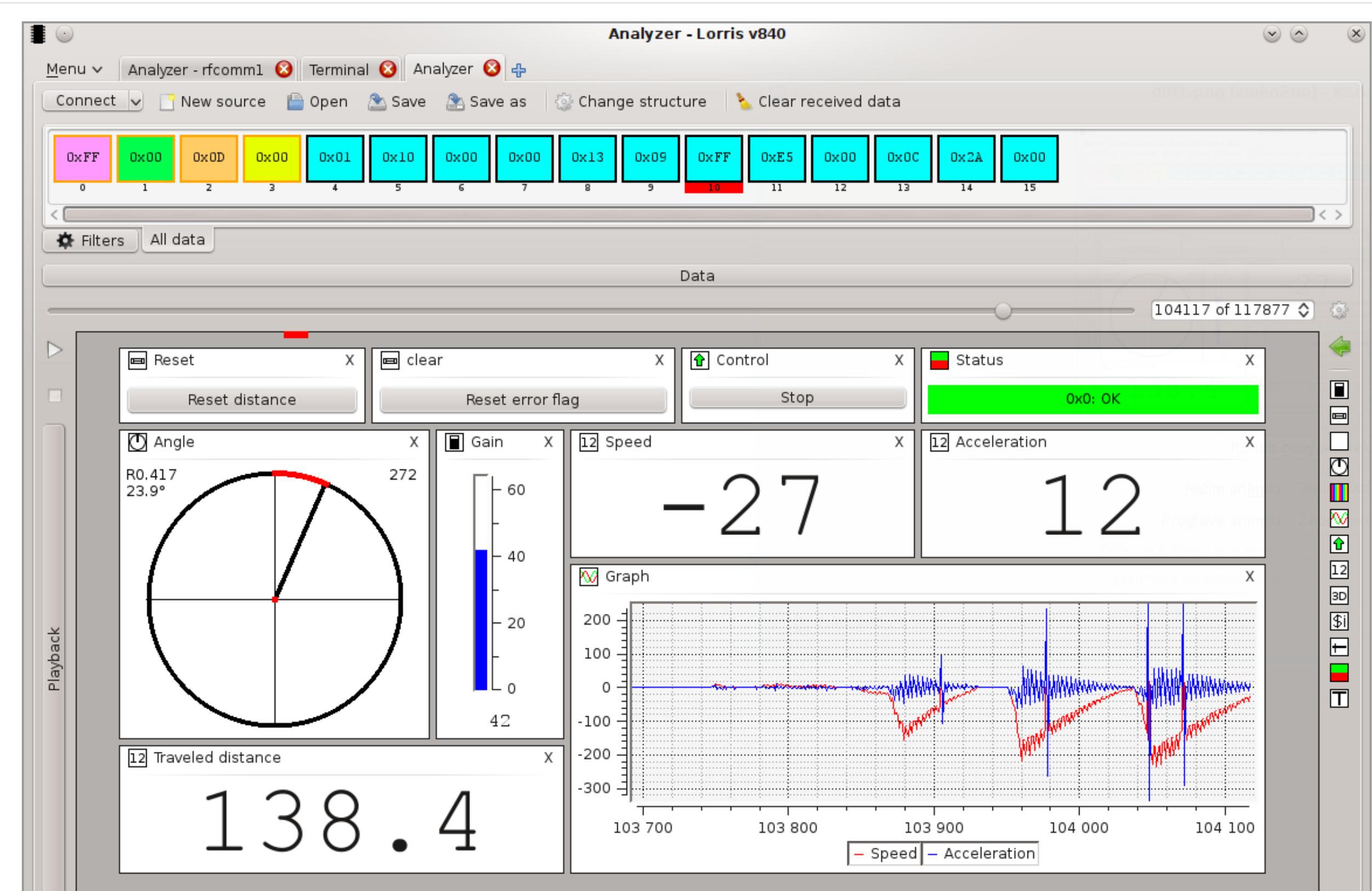
Lorris mobile



Lorris mobile is a portable addition to the desktop program. It is application for Google Android which contains the most important tools - programmer, terminal and joystick, others are currently in development. Each tool is opened in a tab, similar to the desktop version. Opened tabs, their content and their settings are saved to sessions.

Bluetooth and TCP socket (Wi-Fi) can both be used to connect to the device. The app requires Android 2.3 or higher.

Lorris' main window



Example UI layout for robot's sensors

This range meter is glowing red because the measured distance is lower than 35 cm. Robot stops to avoid crashing into something.

Widget Color: data from the color sensor. Numbers are the RGB values.

Widget Number: Range measured by ultrasound range meter mounted on top of the robot. The displayed value is in centimetres.

Widget Status: displays state of the robot, e.g. if there is some obstacle in the way.

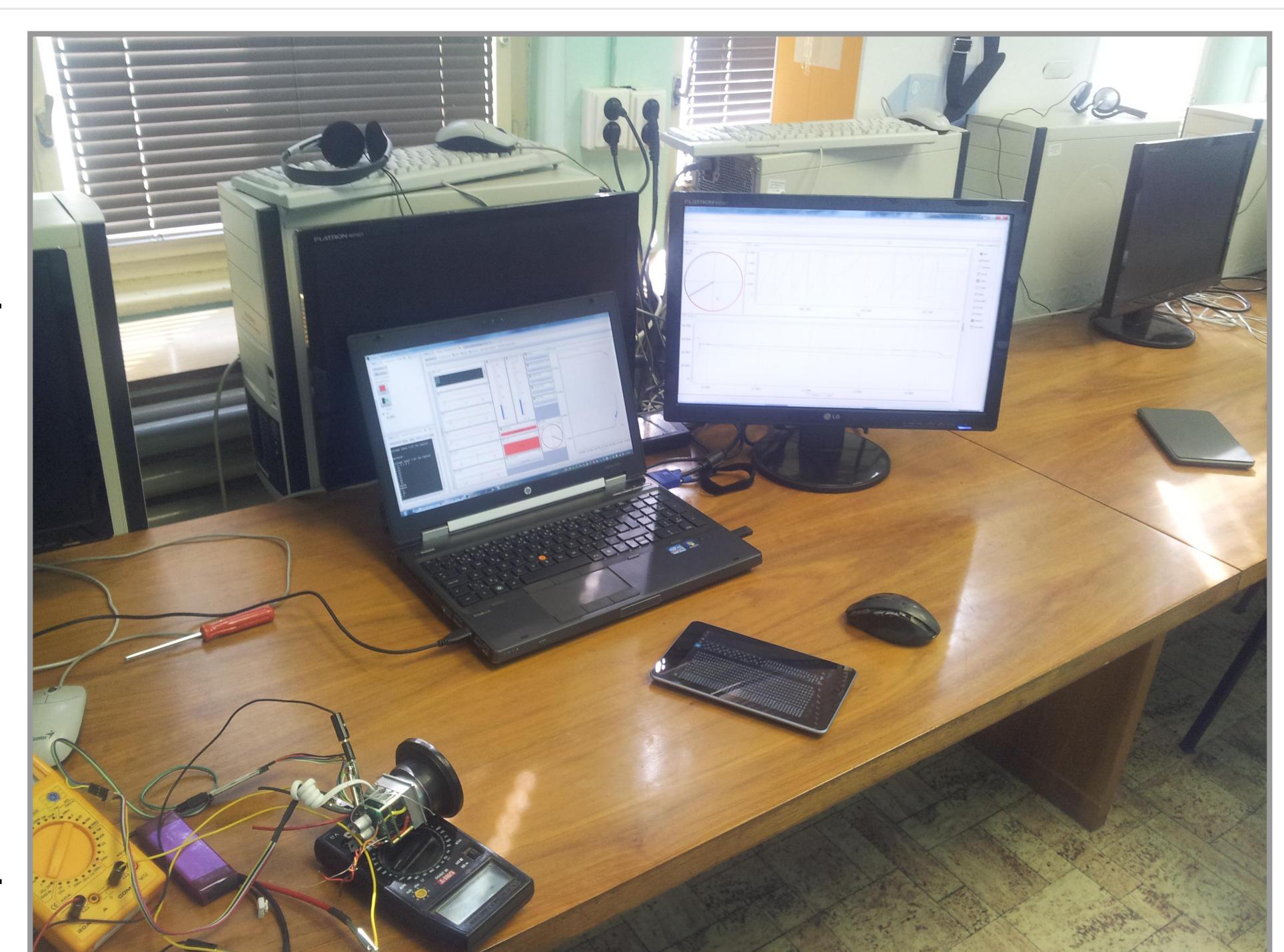
Widget Number: two encoders measure travelled distance. In this picture, the robot has been moving backwards and slightly right.

This button is inside the robot and is pressed if robot currently has the game element inside.

Widget Color: there is a button in each corner to detect collisions. Green means the button is currently pressed.

Real-world use

Several companies have already been using Lorris for more than a year, among them is children and youth centre Junior in Brno, Czech Republic, with its technical-oriented classes. Many robots to compete in student competitions (like Robotic day in Prague) are made there and Lorris significantly helps with their development. This gives me live feedback from several dozen real users, for whom Lorris became irreplaceable development tool.



Value of Lorris as a tool for developers equals that of 3D CAD software for engineers. It significantly simplifies and speeds up the whole process of development applications for microprocessors. In addition to that, it even enables new methods of development which were not possible because of lack of proper tools. For example, thanks to scripts support, it is possible to simulate the whole device inside the analyzer.