EMERTXE TRAINING PROJECT DOCUMENTATION FRAMEWORK

REQUIREMENTS & DESIGN DOCUMENT

Module – Qt Programming

Car Digital Dashboard

Contents

1 Abstract

The goal of this project is to provide participants a real-time hands-on exposure in creating a car digital dashboard. Given that automotive are getting smarter, this project will help you to understand exporting all critical data to the user so that effective decisions can be made. It will help you to create compact, reliable and effective solution for digital car dashboard. This involves car sensor simulator and graphical live status. Using Qt control window all possible car environment will be simulated. Graphic live status must update real time status and should be independent to integrate with real sensor input easily.

There was a time when the meaning of automobile was mechanics. Then slowly electronics also started playing big roles in automobiles. And today it means mechanics and electronics. Modern car is a sophisticated unity of high-accuracy mechanics and high-end electronics. Today microprocessors, those creatures of science world, are everywhere. Modern automotive microprocessors are highly integrated hybrid systems, that contain central processing unit (CPU), graphic processing unit (GPU) and additional peripheral for interacting with other chips, units and sensors like CAN, I2C and UART interfaces. Everybody who likes cars and driving, the first picture when mentioning car is wheel and dashboard. A well-shaped wheel, curved lines of torpedo, rows of gauges, switches and buttons. Now think about all these are coming to touch screen LCD, a digital dashboard.

2 Requirements

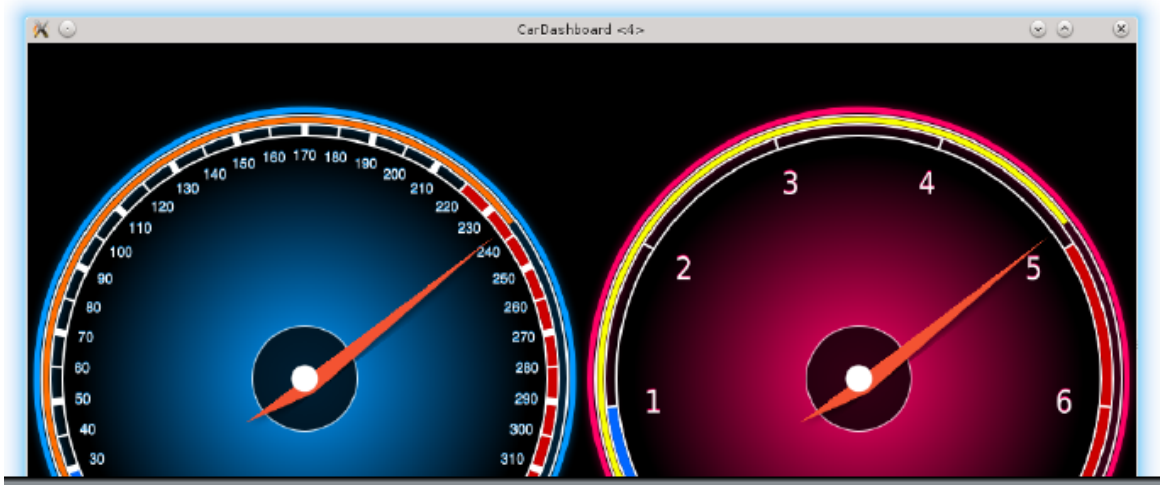
The aim of this project is to create compact, reliable and effective solution for digital car dashboard. Virtual instrument cluster development includes two major fields: car sensor simulator and graphical live status.

Car sensor simulator

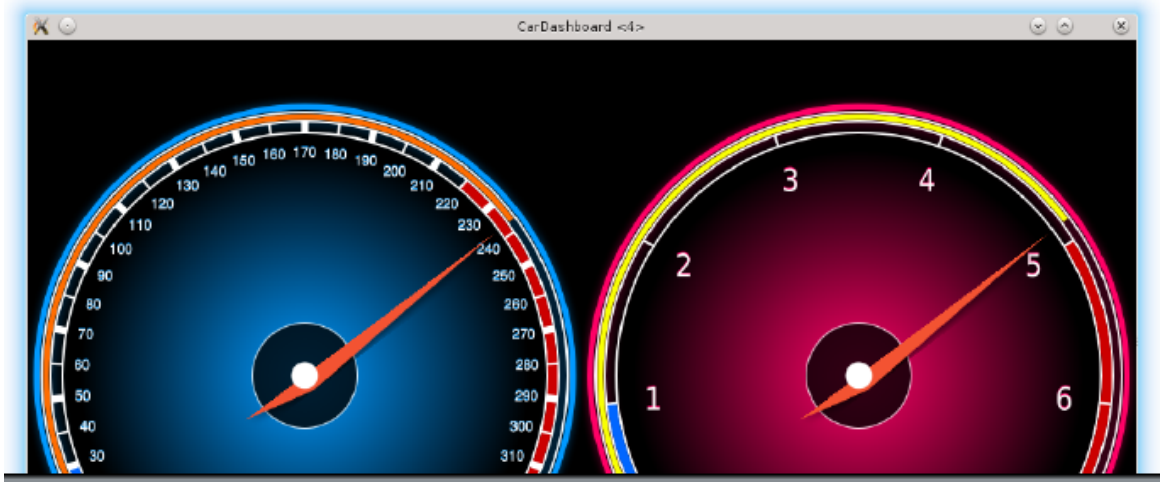
In this we have to simulate all possible car environment using Qt control window. In this project we are trying to simulate following.

- Speedometer.
- Trip meter (Optional).
- Tachometer.
- Odometer.
- Fuel gauge.
- Gear position indicator.
- Engine Coolant temperature.
- Oil temperature and pressure.
- Front seat-belt indicator.
- A/C status.
- Right/Left/Straight indicators.
- Wiper with speed.
- Door lock indicator.
- Door window open/closed status(Optional).
- Low/High/Parking light indicator.

GUI requirements



3 Sample Output



4 Artifacts

4.1 References

- Qt - Automotive Suite
- Qt for Embedded Systems
- Qt based GUI for Raspberry Pi