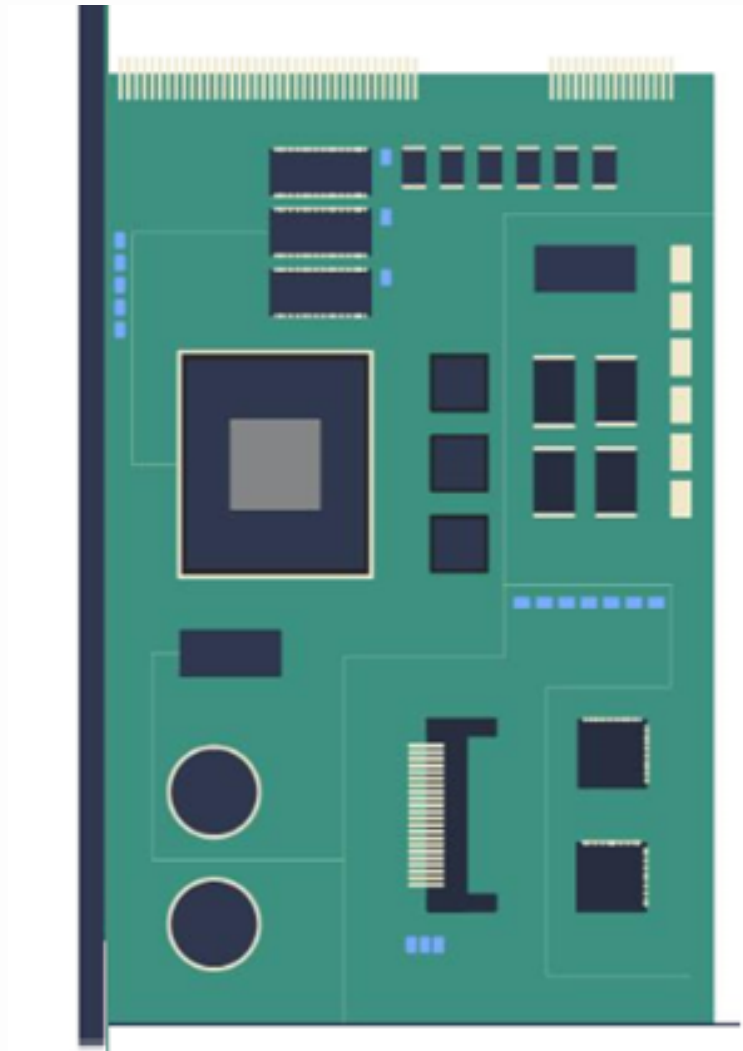# APPLICATION DESIGNER FOR THE RAIL INTERLOCKING CONTROL

Datasheet

Project objective

Create a tool for the development of the source code for microprocessor-based controller applications to provide basic monitoring and control of interlocking operations. Applications are source code files of a board-specific language for implementing execution logic, which become files of a special format after compilation and are freely uploaded into and executed by microprocessor-based interlocking controllers.

Result

The Application Designer is used both by our engineers to continue delivering interlocking solutions for the customer, and by all the developers who create interlocking solutions using our customer's equipment. After the introduction of the tool the delivery of interlocking solutions has accelerated by 50% while human errors in the design are eliminated.

Scope of work

- IDE software development for creating applications to execute rail interlocking operations
- Error checking function – displaying a warning sign instead of the line number to the left and a red squiggly line under the syntactically incorrect code
- Syntax highlighting function to displays source code in different colors and fonts according to the category of terms
- Logic design panel to visualize relay equivalent circuits based on the application code. Users can see the switches, coils, and bits associated with any application file
- Hardware design panel to build and visualize the hardware architecture of the application file. Users can create a visual image of the interlocking devices with card files and boards, and create applications not by writing code but by using visual tools
- Model panels for re-use of templates with code blocks for different objects of the same type within the projects. Models are user-defined data types that act as the blueprint for individual objects, attributes and methods in object-oriented programming – used for the description of physical objects that can use bits assigned to the controller

Activities

- Requirements definition
- Architecture design
- Software development
- UI development
- Writing technical documentation
- Product testing
- User manuals and video tutorials creation
- Support activities

About the project

Technologies

- Java
- PowerShell
- Python
- Unittest
- Docker
- GitLab
- PyTest
- Pywinauto
- JUnit

Project size

- 1 Technical Coordinator
- 5 Software Engineers
- 2 QA Engineers
- 1 Project Manager
- 1 Graphic Designer
- 1 Technical Writer

Duration

32+

32+ months
from April 2020

Platform

Windows