

Refurbishment Tool For Telia Company

High throughput production test system

Fixed development cost

Easy to upgrade



Refurbishment tool for Telia Company

Telia Company AB offers wireless logging units for cars that can revolutionize car insurance. By logging the driving pattern of the driver, insurance companies can offer truly individual insurance policies. Careful drivers no longer need to subsidize reckless drivers. If a customer terminates their insurance, the logging unit shall be refurbished, meaning previous data should be cleared and the unit should be reinstalled, possibly even upgraded and tested, so that it can be reused again. The development of such a refurbishment tool presents a number of challenges. Telia Company hired Novator Solutions to address these challenges.





Challenge

Solution

Technical

- Multiple buses and protocols
- Non-standard protocols
- High throughput, user friendliness and operator training
- Need for low maintenance and possibility of upgrading

- High-end and modular commercial-off-the-shelf hardware with reconfigurable FPGA
- Parallel of multiple units in the same highly automated refurbishment test software based on industry standard tool NI TestStand
- Hardware abstraction layer architecture to allow for future upgrades

Business

- Hardware testing outside of core business
- Limited in-house expertise
- Need to support multiple device vendors with the same platform and refurbishment process
- Short time schedule
- Needed to avoid financial risk
- Multiple stakeholders
- CE marking and documentation

- Full responsibility for project management, development and supplier communication
- Fixed price project
- Full responsibility for CE-marking and documentation

Device under test

The device under test (DUT) is an on-board diagnostic device with an integrated 4G modem that continuously transmits the driving data to the insurance company. Its electrical signalling protocols and pinouts are defined in the OBD-II standard. Different car manufacturers use different subsets of the available protocols and buses, which means that the devices need to support a vast variety of

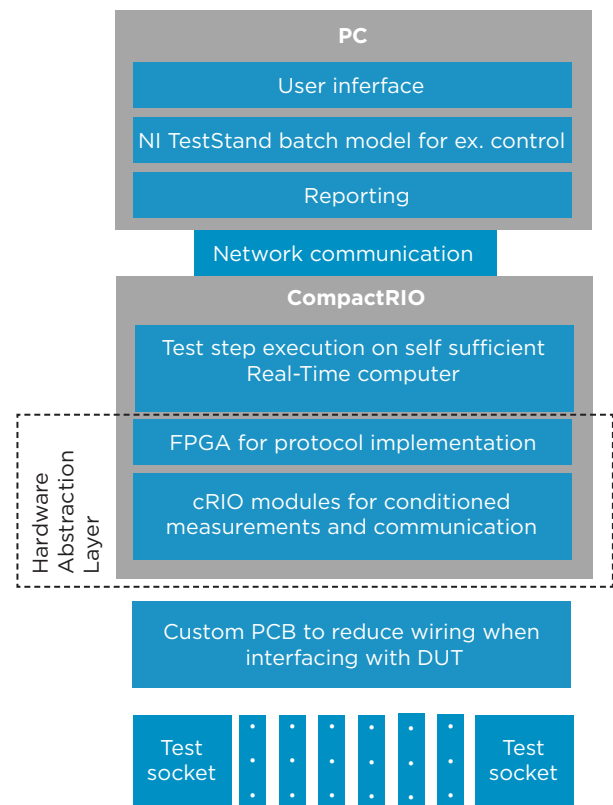
combinations of pinouts and protocols. During a complete refurbishment process the function of all of these needs to be tested. The test system needs to support communication of standard protocols, such as CAN, J1850 VPW, J1850 PWM, and ISO9141, as well as a custom protocol for controlling the DUT and downloading new firmwares.



Test system architecture

The hardware interfaces are implemented on a NI CompactRIO, and a network communication layer is used to establish communication with a PC running a test sequence developed in NI TestStand. TestStand is a widely used test executive and framework for common tasks within a production test environment, which handles everything from sequencing of test steps to report generation and cloud database integration.

By separating the DUT specific features of the test system from common tasks within a test environment, a lot of the functionality in the NI TestStand environment was leveraged to provide a complete and proven solution for parallel testing of multiple devices.



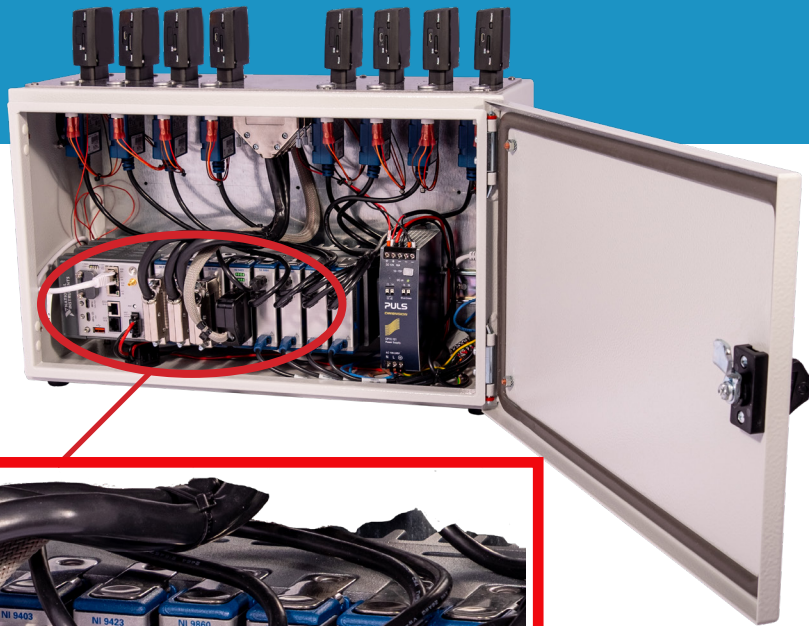
Hardware and test socket abstraction

A driver layer was developed to handle the low level communication with the DUT and an object oriented hardware abstraction layer (HAL) was developed to map each test socket with the associated pins on the IO modules. Each test socket is associated with a separate set of driver object instances to allow independent execution on any one of the initialized test sockets.

The protocols were implemented on an FPGA which handles input and output buffers as well as sending and receiving the physical signals.

The implementation is hardware timed and features parallel implementations of all communication protocols and IO buffers. To reduce wiring, a custom designed PCB was created for the physical interface with the DUT.

This architecture enables flexible execution of test steps and as the lower layers are generalized and abstracted, the system can be updated to handle new types of OBD-interfaces with minimal modifications.



Hardware platform

The CompactRIO platform encapsulates a Real-Time processor and an FPGA, both programmed with LabVIEW, as well as a broad range of IO-modules.

The Real-Time processor handles the network communication and execution of the test steps ordered by the test executive.

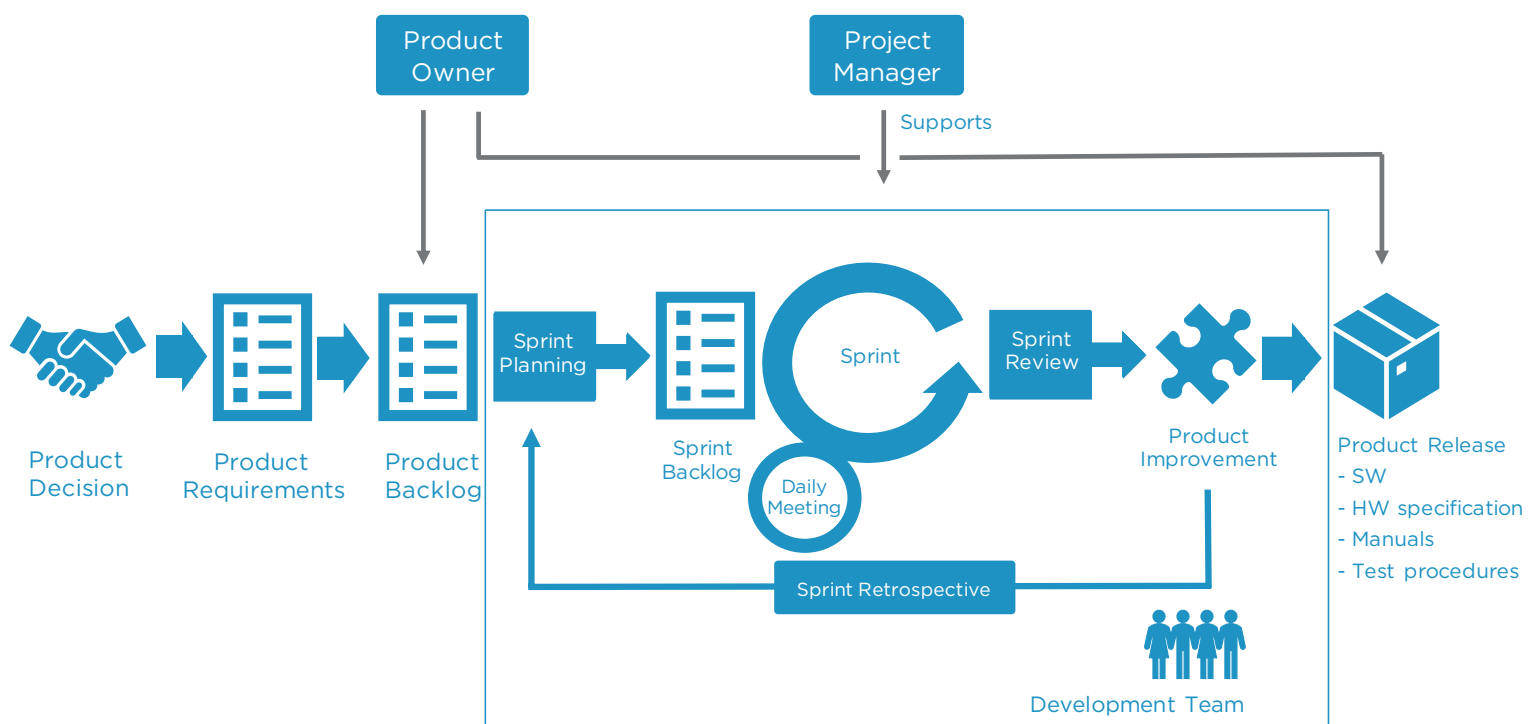
The FPGA provides flexibility to meet the needs for all the protocol implementations, allowing rapid development of an interface to communicate with the DUT using a vendor defined non-standard protocol, e.g. to upload firmware to the DUT.

The range of off-the-shelf-available IO-modules simplified the system design process and reduced the technical risk of the project. The only PCB development needed was the adapter board to interface with the DUTs through standardized cabling and connectors, which was used mainly to avoid excessive cabling.

In addition to speeding up the development process, the use of commercial hardware simplifies future maintenance and replication offering a very high level of flexibility and functions as a platform to be expanded when new DUTs are introduced.

Novator Solutions' development process

Novator Solutions uses the Scrum process framework for managing our development work. Scrum is a lightweight, iterative and incremental framework for managing complex work. The framework challenges assumptions of the traditional, sequential approach to product development, and enables teams to self-organize by encouraging physical co-location or close online collaboration of all team members, as well as daily face-to-face communication among all team members and disciplines involved.



For our customers, this ensures efficient prioritization and straight forward communication as the project proceeds. All to ensure the highest possible customer satisfaction with the end result, as well as the most cost-efficient way of reaching the desired solution.

In the development of complete systems, we can take full responsibility for the entire development chain: From compiling the list of demands through design and construction of mechanics, pneumatic and electronics, to software and database development and installation. In order to maintain the highest level of all parts of each system, we always use trusted and certified partners in areas beyond our own expertise.

Novator Solutions' pricing models

Novator Solutions offers two types of pricing models for turn-key systems, Fixed Price or Time & Materials. Development is always done using the agile programming framework Scrum.

Fixed Price

In a fixed-price-project the deliverables and how to evaluate them are decided before the project starts. If needed Novator Solutions can perform a pre-study to help the customer identify the requirements.

Once we have the requirements, we provide a quotation with an all-inclusive price for a system that fulfills them, including factory and site acceptance tests to be performed to verify the functionality upon delivery. A product owner is appointed by Novator Solutions and we develop the system fulfilling the already agreed upon requirements.

Pros

- No financial risk – customer knows from day one what they will pay.

Cons

- Limited possibility to reprioritize deliverables after order – deliverables must be thoroughly validated before project start.
- Limited possibility of beta-testing or pre-releases. We work according to requirements document and test programs until full functionality can be delivered.
- Commitment to pay full system delivery even if certain functionality is reprioritized.

Time & Materials

In a Time & Materials project, a scope for the project is set. A list of deliverables and preferably how to evaluate them are often decided on, and Novator Solutions give an estimate on the time we think it will take to implement the system.

A product owner is appointed by the customer, and a project manager by Novator Solutions. They stay in close contact throughout the project and meet up with regular intervals during which the deliverables list can be reprioritized, amended or deducted to achieve the best possible system with the least amount of investment.

Pros

- Validation work of deliverables can be less thorough, which saves time.
- Customer can test beta-versions of system and change deliverables to optimize it throughout the project.
- Project can be closed as soon as the desired functionality level has been achieved. Or a first versions can be deployed while we continue to work on upgrades.

Cons

- Customer has to take the financial risk of a decided upon termination period.

Regardless of delivery model chosen, we are always committed to the long term success of the systems we deliver. We offer service agreements to guarantee uptime and continuous upgrades, and to mitigate obsolescence of components.

About Novator Solutions

Novator Solutions, based in Stockholm, Sweden, is specialized in developing highly automated and flexible test, measurement and control systems based on NI platforms such as CompactRIO and PXI. We offer turn-key solutions that help companies with electromechanical products raise their product quality in a cost-efficient way that scales with their future needs.

Novator Solutions is a NI Gold Alliance Partner, as well as winner of numerous awards including Alliance Partner of the Year 2015 and 2017 in northern Europe and Alliance Partner of the Year 2016 in Europe, Middle East, India and Africa Region (EMEIA).

Our Partner

NI is an American company based in Austin, Texas with over 7.000 employees worldwide. NI provides an integrated hardware and software platform that helps engineers and scientists in nearly every industry. NI products and solutions span from design to test. Its high-performance PXI platform deliver unprecedented flexibility, accuracy, and measurement speed, and its robust CompactRIO platform provide high-performance processing capabilities, sensor-specific conditioned I/O, and a closely integrated software toolchain in a small mechanical form factor, making it ideal for Industrial Internet of Things (IIoT), monitoring, and control applications.

