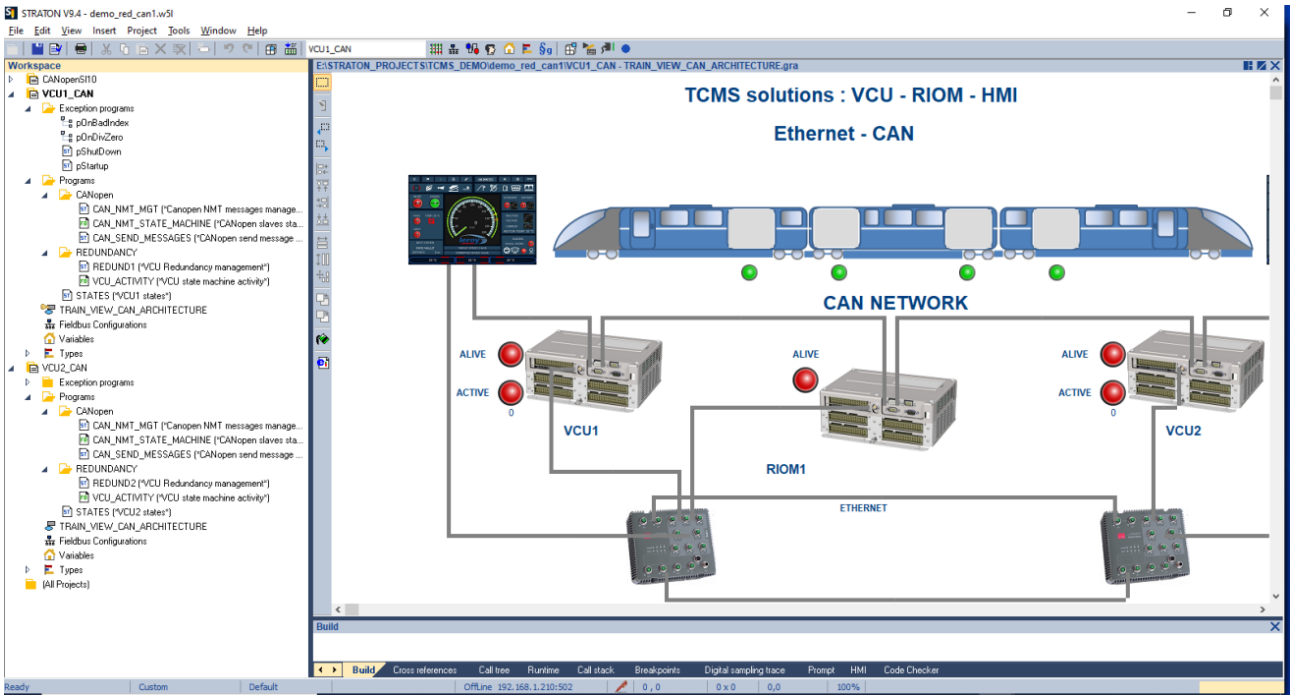
Redundant train control and monitoring system (TCMS)

Leroy Automation, part of Agon Electronics, has developed a redundant TCMS host controller with CANopen NMT (network management) manager functionality. The product is suitable for mainlines and metro cars as well as power sub-station automation with dedicated remote terminal units (RTU).



(Source: Leroy)

The advantages of CAN lower layers and CANopen higher-layer protocols are key elements in the success of this project. This includes the robust and reliable communication regarding the lower layers and the off-the-shelf interoperability of the higher layers. The availability of CAN hardware and CANopen software as well as tools for reasonable prices was another selection criteria.

Unlike, historical buses and protocols based on EIA-232 or EIA-422/-485 such as Modbus-RTU and Profibus, or even MVB (Multifunction Vehicle Bus), CAN-based networks allow the transmission of prioritized data frames without altering or destroying the data frame currently sent. Indeed, EIA-232 and EIA-422/-485 serial buses allow commander/responder polling-type protocols, where the commander "speaks" and the responders "listen" by responding each one at a time. This method, called "polling", requires a cycle whose duration depends on the number of responders and the quantity of data to be transmitted.

On the contrary, CAN-based networks allow communication according to the producer/consumer model, where each node, whether commander or responder, can speak whenever it wants, without worrying about the state of the network. This method achieves a much higher performance, because frame collisions are taken care of by the CAN controller, which facilitates the developer's work.

The CAN CC (classic) protocol covers the data link layer and the physical coding sublayer of the 7-layer OSI (open system interconnection) model. EIA-232 and EIA-422/-485 specifications relate to the physical layer. They only provide basic rules and the physical link for byte exchange to enable transmission of serial data using a multi-drop bus topology. The content of the frames is user-defined. This is an advantage in terms of simplicity, but when it comes to compatibility, it becomes a disadvantage given the multitude of existing protocols.

CANopen is a high-level protocol (layer 3 to layer 7 of the OSI model), which implements the mechanisms necessary for monitoring the network, transmitting not-confirmed PDO (process data object) messages and confirmed SDO (service data object) messages. CANopen thus supports commander/responder-type communication (NMT messages), producer/consumer-type communication (PDO), and client/server-type communication (SDO messages).

With asynchronous EIA-232/-422/-485-type links, the designer needs to develop a proprietary protocol, which ▷

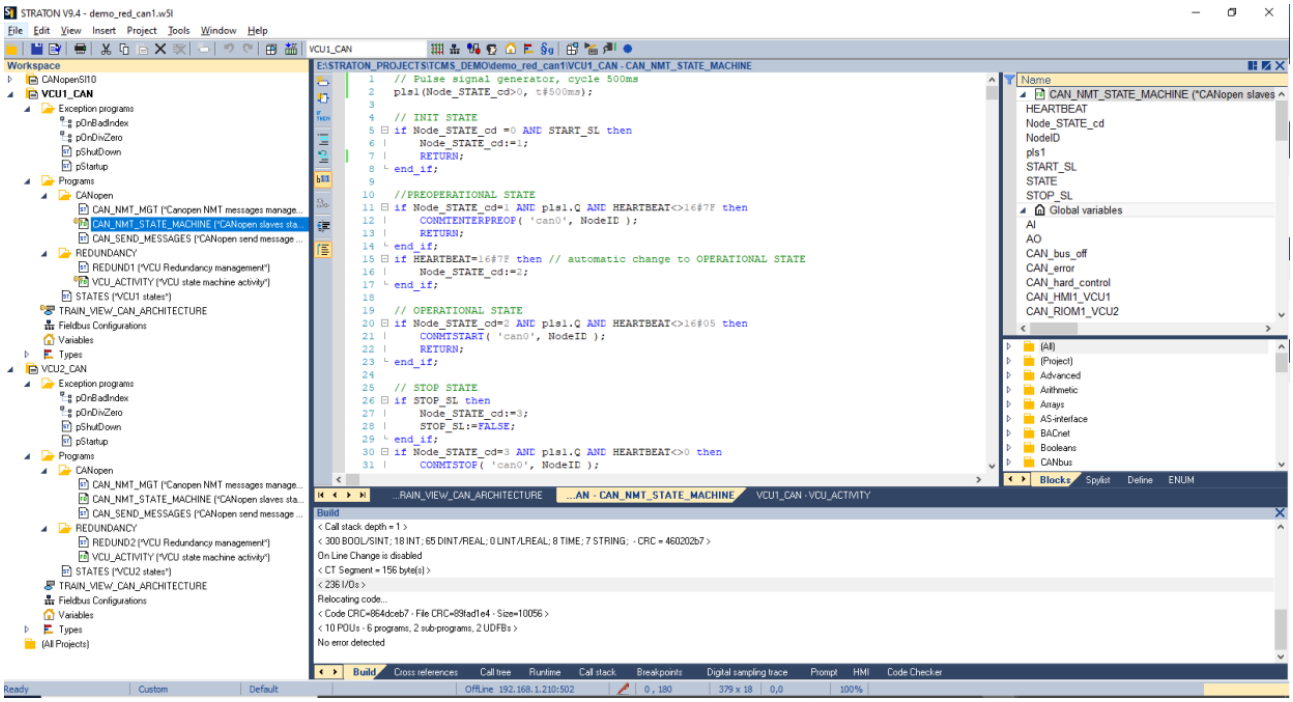


Figure 1: Screen of the Straton programming and configuration tool (Source: Leroy Automation)

requires significant time for definition and implementation, with support for each node of the network. The development may take a significant time and the reliability and robustness of a proprietary protocol could be uncertain!

The CANopen object dictionary

A fundamental notion of CANopen is the object dictionary. It is used to inform other CANopen devices (normally the unit with NMT manager functionality) about the properties of a device, in a standardized manner by means of SDO communication. This mechanism thus allows this device to be modeled, in order to make the hardware independent of the control software. The benefit for the end user is to be able to choose their equipment independently of the manufacturer.

In order to facilitate this approach, significant work has been carried out by the non-profit organization CiA (CAN in Automation) through specifying CANopen profiles: generic I/O modules (CiA 401), drives and motion control (CiA 402), medical devices (CiA 412), control systems for elevators (CiA 417), and profiles for rolling stock specific equipment:

- CiA 421: CANopen application profile for train vehicle control networks;
- CiA 423: CANopen application profile for rail vehicle power drive systems;
- CiA 424: CANopen application profile for rail vehicle door control systems;
- CiA 426: CANopen application profile for rail vehicle exterior lighting control;
- CiA 430: CANopen application profile for rail vehicle auxiliary operating systems;
- CiA 433: CANopen application profile for rail vehicle interior lighting control.

The CANopen connectable TCMS controller and the RIOM product line by Leroy is an embedded control system with built-in input/output features designed to be on-board integrated in rolling stock vehicles. The products are compliant with the EN 50155 standard. The VCU (vehicle control unit) TCMS devices feature CANopen NMT manager redundancy. They are on the same CANopen network negotiating who is the active NMT manager and taking over the NMT responsibility in case of a malfunction of the active NMT manager. For maintenance purposes an additional Ethernet link is applied.

VCU programming and redundancy

The host PC (personal computer) workbench used for programming the VCU is the Straton tool from Copa-Data. It comprises:

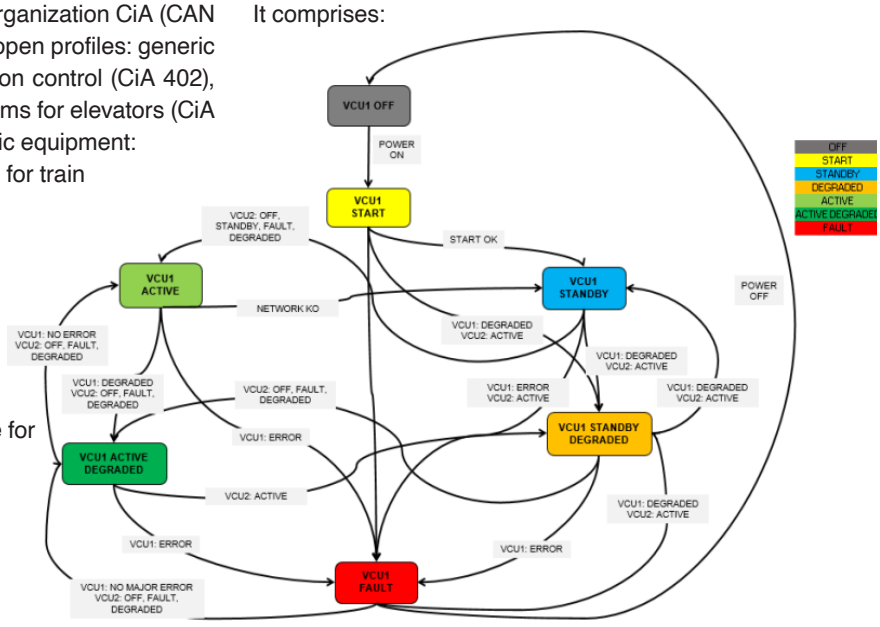


Figure 2: VCU "hot" redundancy state diagram (Source: Leroy Automation)

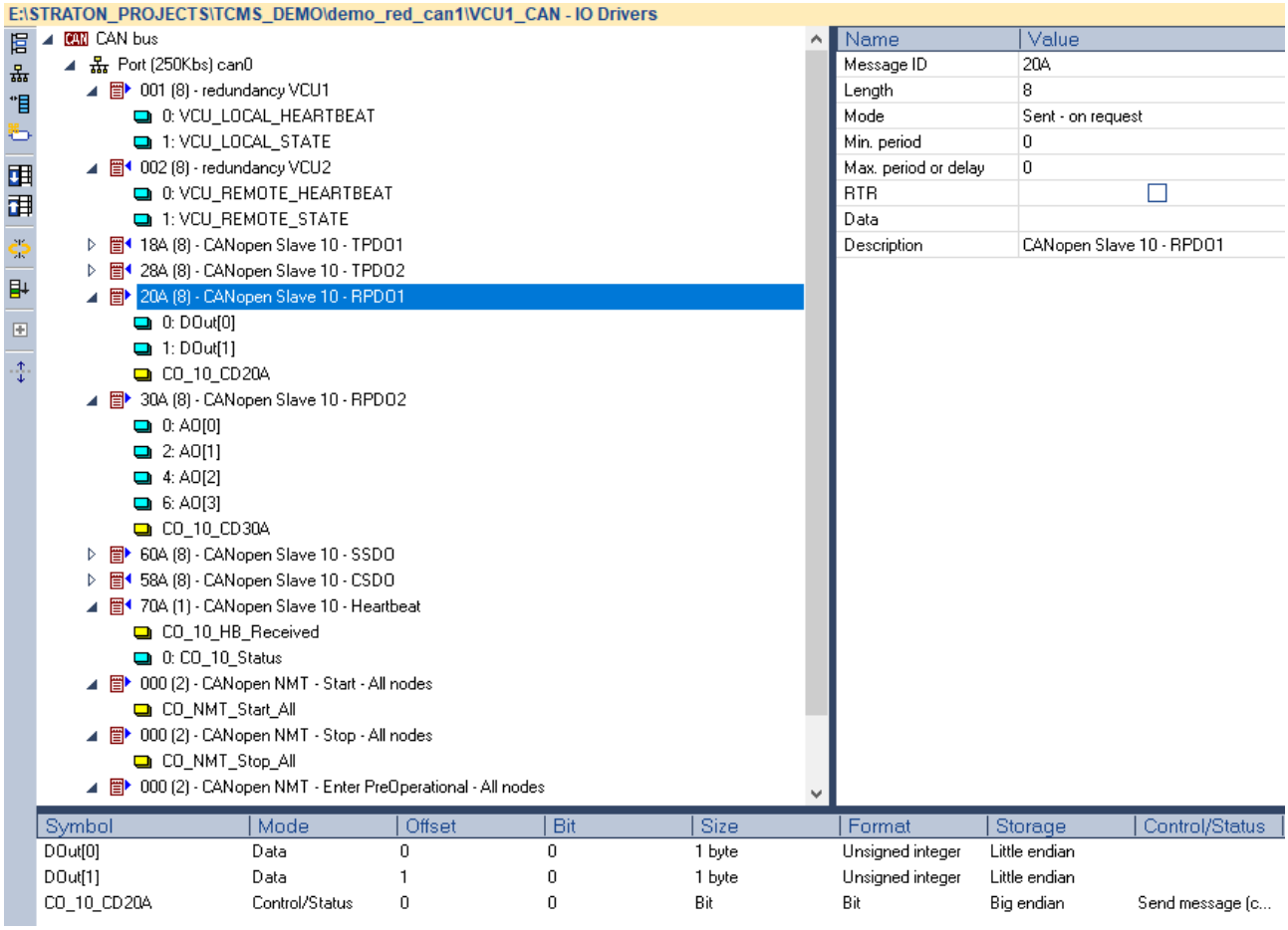


Figure 3: Configuring PDOs with the Straton tool (Source: Leroy Automation)

- an IEC 61131-3 languages programming tool;
- a CANopen editor for network configuration;
- real-time monitoring tools for debugging purposes.

The tool shows in Figure 1 a list of projects in the upper workspace area. The three available projects can be designed and debugged. The project called “VCU1_CAN” (see Figure 1) is divided into several programs and tools to manage different features: startup, VCU redundancy as well as CAN and CANopen communication.

The VCU redundancy is managed to enable a “hot” redundancy, if one VCU fails. In each VCU, two programs manage the redundancy process:

- Redundancy management with the second VCU (“REDUND” program): This program calls the function block “VCU_ACTIVITY”.
- VCU activity state machine (“VCU_ACTIVITY” function block): This function block allows to calculate the VCU state: start, standby (nominal), standby (degraded), active (nominal), and active (degraded).

CANopen communication

The CAN communication is determined by means of the configuration editor and structured programs. The CANopen messages are configured by means of the Straton tool. Data exchange and diagnostic/control variables can be defined for each CANopen message. CANopen messages mapped to CAN data frames using CAN-ID 1 and CAN-ID 2 are used for the VCU redundancy between VCU 1 and VCU 2. Each VCU sends a heartbeat message and its VCU activity state: those values are used in the “REDUND” program to manage the VCU activity state. PDO messages mapped to CAN data frames with CAN-ID $18A_h$ and CAN-ID $28A_h$ are transmitted by the CANopen device with the node-ID 10. The PDOs mapped to CAN data frames using CAN-ID $20A_h$ and CAN-ID $30A_h$ are transmitted by the active VCU to the CANopen device with the node-ID 10. These PDOs set the message control parameter respectively the “CO_10_CD10A” parameter.

Several programs have been developed to manage the messages defined in the fieldbus configuration editor:

- PDO transmitting program CAN_SEND_MESSAGES: This program is used to trigger the sending of PDO messages from the active VCU host controller to other CANopen devices with NMT server functionality.
- NMT program “REDUND”: Only the active VCU sends NMT messages and manages other CANopen devices. It calls the CANopen state machine function block instance for each managed CANopen device.
- CANopen state machine function block CAN_NMT_STATE_MACHINE: This function block allows to manage the CANopen states for a CANopen device.

For debugging the system, several Straton tools are available: list of variables, graphic monitoring views, soft oscilloscope, test sequences, output-window view, and digital-sampling traces. In a graphic-monitoring view, all parameters can be animated by the project variables’ values, when in simulation or online mode.

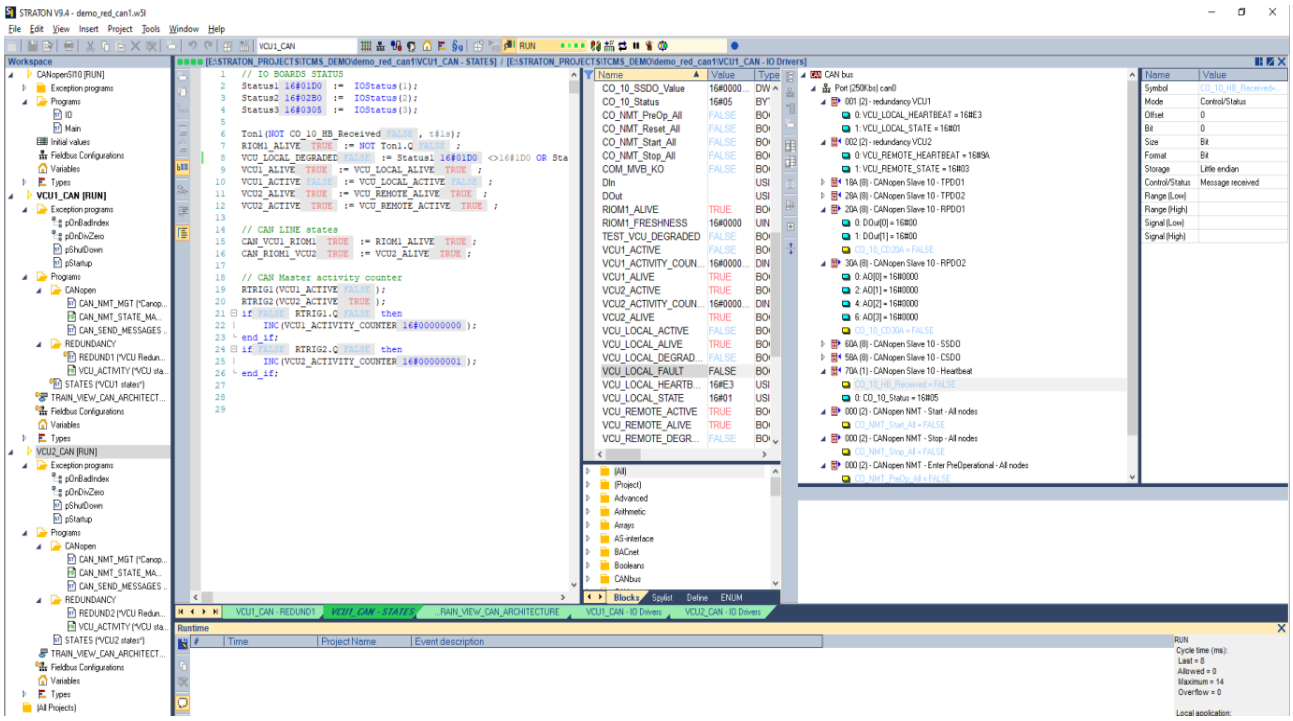


Figure 4: Busmaster software tool (Source: Leroy Automation)

The three devices in the system were downloaded with their Straton project from the project list. All of those devices can be debugged at the same time through the workbench or in several workbench instances. The low-level CAN communication can be monitored via a PC with a specific CAN network software tool Busmaster. An USB/CAN dongle is applied to connect the PC to the CAN network. The tool can monitor all CAN frames and the bus load.

hz