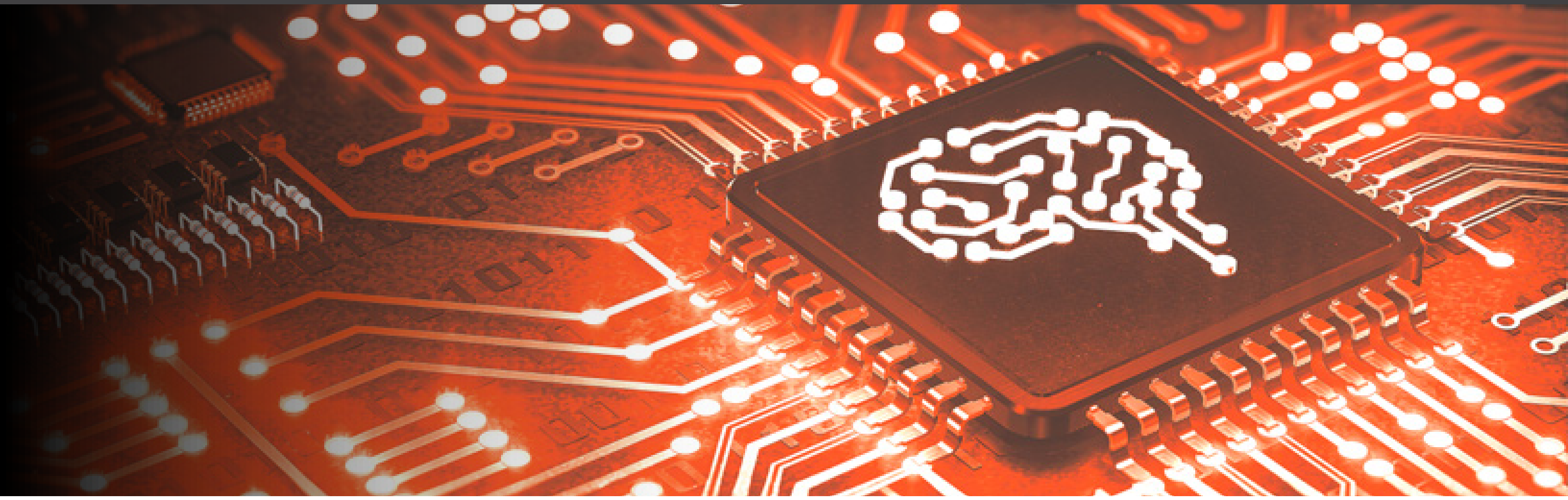


Central Vehicle Controllers Make Software-Defined Vehicles Possible

Software-defined vehicles enable innovation in mobility unlike any technology the automotive industry has ever seen, with new functions and capabilities just a wireless download away. But something has to connect the software with the vehicle to turn engineers' dreams into physical reality.

The central vehicle controller (CVC) fulfills that role. The CVC can be a power and body controller, propulsion and chassis controller, data network router, gateway, firewall, zone master and data storage hub all rolled into one – or it can perform a mix of some of those functions. More importantly, it is a key piece of the architecture that translates software code into physical action, from bits and bytes into mobility itself.

The CVC handles the details of communicating signals with hundreds of components in the vehicle and then can help abstract those functions as services to software applications. It allows developers to spend less time worrying about how to handle communications within a vehicle and more time creating features that add real value for consumers. Without the CVC, the software-defined vehicle does not exist.

THE ‘LITTLE BRAIN’

A runner settles into the starting blocks, ready for the race to begin. When the sound from the starter horn reaches the runner’s ears, the brain tells the legs to push hard against the ground to propel the body forward. But the runner is thinking about speed and focus, about distance and pacing, and maybe about advice from her trainer. She is not thinking about which neurons to fire to make her muscles contract in the correct sequence, and she is not thinking about how to maintain balance as she rounds the final turn toward the finish line.

The cerebrum handles the higher-level thinking, while the cerebellum (the “little brain”) coordinates the muscles to carry out the actions ordered by the cerebrum.

This analogy is particularly apt as we move toward software-defined vehicles. There is a need for a layer in between the brain and nervous system of a vehicle: a “little brain” that bridges the digital world with the analog, translating the decisions made by the brain into actions carried out by the nervous system of the vehicle quickly and efficiently.

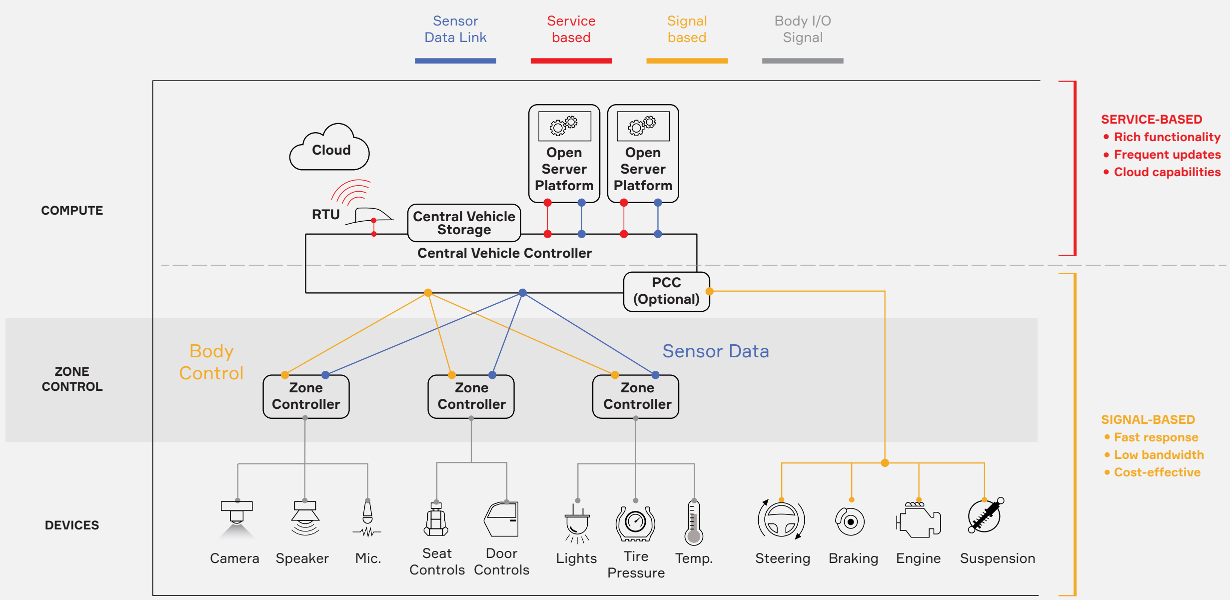
This is the central vehicle controller (CVC), a specialized computer that plays a pivotal role in tomorrow’s vehicle architectures.

SIGNAL TO SERVICE

To make the translation happen, the CVC bridges the signal world to the service world. Vehicles have long been governed by signals, where a

The Central Hub

The central vehicle controller manages communications throughout a vehicle, bridging the service world of software applications and cloud computing with the signal world understood by I/O devices.



body controller sends a data frame to a specific device, which then reacts by performing a specific function. In a service-oriented architecture, functions are presented as more generic services instead, and any apps that require those services subscribe to them.

For example, an OEM may want to allow users to control vehicle climate through an app in its infotainment system. The CVC would present a service called “HVAC,” and the app would subscribe to the HVAC service, communicating with it through application programming interfaces (APIs).

When the user selects a temperature, the climate app in the infotainment system would use those APIs to tell the HVAC service to set the temperature to, say, 20 degrees Celsius for the right side of the vehicle. In turn, the HVAC service would send signals to the various actuators — to the compressor, the fans and the flaps — via zone controllers. The service would figure out when to turn the compressor on or off, which fans to activate, and how far to open the flaps and then send signals to the appropriate devices at the appropriate times. Where it makes sense, the services could focus on very narrow functions, or microservices, allowing for updates that are targeted to just those functions.

Through this abstraction, the infotainment system app in this example is completely insulated from the mechanics of climate control. It is not aware of the location of the climate control mechanisms or even of their existence. Because the developers of the infotainment system do not have to worry about those mechanics, they can instead focus on optimizing the user experience — and improving how they use climate control in various app innovations.

DIFFERENT TIMELINES

The separation in this architecture puts functions that do not change as frequently onto a different platform from those that could potentially update often over time. The CVC might house functions that are on the same innovation cycle — every five or six years — as the vehicle itself. This would free apps, which might be housed in an open server platform (OSP), to change as often as needed through over-the-air updates. In addition, OEMs could upgrade their OSPs as more powerful microchips emerge — say, every two years — which would put those higher-level functions on an upgrade cycle that was closer to that of today’s smartphones.

> *“With its real-time focus and connections to devices throughout a vehicle, the CVC acts as a router for all data communications. It handles the prioritization and scheduling of network traffic.”*

With that distinction in mind, the CVC becomes a natural location for many power and body control functions and optionally for the propulsion and chassis controller (PCC).

The power and body controller manages all of the devices related to the body of the vehicle, including interior and exterior lights, window controls and door locks, climate controls, warning lights and overall power distribution. These functions are less safety-critical than the PCC but similarly do not change much over time.

The PCC can include the higher-level applications of braking, steering, suspension and engine applications, as well as controls for executing maneuvers as part of an advanced driver-assistance system (ADAS). These functions are built to the most stringent safety requirements and are often rated ASIL-D, which represents the highest level of risk management. Because they are critical for vehicle safety, they have to be approved by regulatory bodies, which can take as long as 18 months — so it is important to keep them off of a platform containing less-critical software that could potentially be updated more often.

This separation provides the best match of processor to function. The CVC focuses on functions that require real-time data processing and a real-time operating system, with data-flow acceleration and high-speed encryption and decryption. In contrast, user experience functions are graphics-oriented, requiring a graphics processor, and complex ADAS functions are more policy-oriented — so those functions would usually run on platforms tailored to those needs.

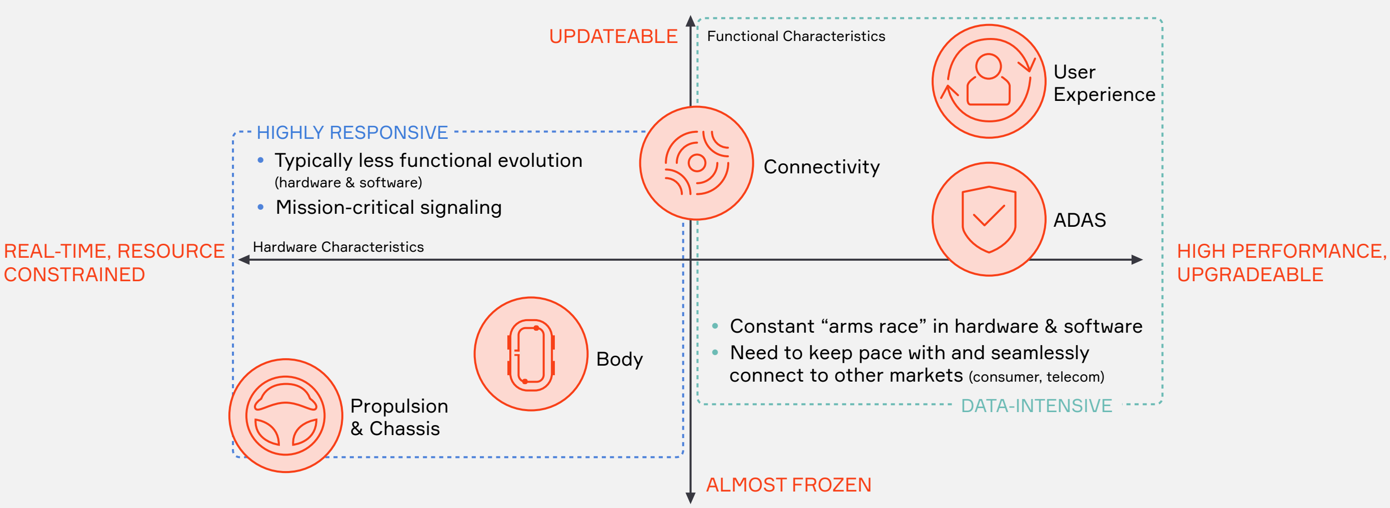
One exception is for entry-level safety features. Vehicles can achieve compliance-focused levels of automation by using smart sensors to preprocess data and send it to the CVC, which could perform sensor fusion to get an accurate picture of the objects around the vehicle, integrate with other systems and execute driving decisions. This approach allows for low-level ADAS functionality without a dedicated domain controller. When the system is developed using Aptiv's next-generation ADAS platform, the platform's additive scalability enables the system to more easily migrate to higher levels of ADAS when required.

THE CENTER OF EVERYTHING

With its real-time focus and connections to devices throughout a vehicle, the CVC acts as a router for all data communications, handling the prioritization and scheduling of network traffic. It is able to route traffic from legacy network technologies such as Controller Area Network and FlexRay with emerging networks such as Automotive Ethernet and PCI Express (PCIe), again insulating higher-level applications from the minutiae of various data protocols.

Different Compute for Different Needs

Not all functions in a vehicle require the same kind of compute or the same frequency of updates



The CVC manages time synchronization, which is critical for the multiple systems within a vehicle to work together effectively. It maintains accurate time through the GPS but can also keep track of the timing if the GPS signal is lost. And on a mixed-criticality network, it ensures that more critical traffic, such as radar data, gets through in a timely manner.

The CVC even manages communications with the cloud. A remote transceiver unit (RTU) includes an antenna and modem to communicate with the cloud via 4G and 5G cellular, as well as Wi-Fi and Bluetooth. The CVC works with the RTU to manage any over-the-air downloads and upload any diagnostics data or analytics to the cloud. This up-integration of processing into the CVC allows for more compact packaging at the point of the antenna.

The CVC can then offer communications through the RTU as a service to software applications in the vehicle. For example, an ADAS software module could subscribe to vehicle-to-everything services — such as traffic-light detection — and the CVC would manage those services.

While cybersecurity is a capability that should be integrated into all of a vehicle's software and hardware, it is particularly important at the CVC, because it serves as the gateway to the outside world. For that reason, it is where the firewall resides. The CVC ensures the integrity of any over-the-air downloads before distributing them to other systems, and it collects any cybersecurity events and reports them to the cloud.

Lastly, the CVC serves as a collection and aggregation point for any data generated by sensors around the vehicle, such as radar, cameras and lidar. The CVC condenses the data, performs sensor fusion and offers information through another service to ADAS applications.

ZONE MASTER

The CVC is a critical component of any advanced vehicle architecture, but it works especially well with zonal architectures. In a zonal architecture, physical areas of the vehicle are segmented into zones, and inputs and outputs (I/O) — sensors, actuators and peripherals — each connect to their local zone controller for power and data. The zone controllers perform some body control functions, preprocess other data and aggregate traffic back to the CVC over a single link, greatly simplifying the data network architecture.

In this scenario, the CVC is the zone master, coordinating actions among all of the zone controllers in the vehicle. Instead of directly connecting 300 to 500 I/Os to a body controller, this architecture separates the I/O connections from the CVC but keeps most body control functionality inside the CVC.

When an app requests that a service turn on the front left blinker, for example, the CVC determines which zone controller to communicate with and sends the signal to blink. The CVC also manages the timing with all of the zone controllers, shielding the app from such details.

Consider the interaction between a cabin camera and an in-cabin user-experience app, such as one that detects objects left behind. In a zonal architecture, the camera might send low-voltage differential signaling frames to a zone controller, which in turn aggregates the data from the camera and other devices onto an Automotive Ethernet link to the CVC. The CVC would then extract the relevant data and, through its services, pass the data to an OSP dedicated to the in-cabin user experience, potentially over PCIe to ensure the highest transfer speed possible. The CVC could simultaneously use the data collected for analytics, processing and sending it via an Ethernet or PCIe connection to the RTU, which in turn would communicate that analytics data via 5G cellular service to the cloud. This would enable the "object left behind" app to efficiently leverage the onboard compute while selectively applying cloud-based object identification or connectivity to notify the user when needed.

THE BIGGER PICTURE

As OEMs plan for next-generation electrical/electronic architectures such as Aptiv's Smart Vehicle Architecture™, the functions of the CVC are a must-have. And as OEMs consolidate electronic control units (ECUs), the CVC remains one of perhaps 10 to 15 boxes in a vehicle. The ECUs could include zone controllers, door nodes, seat nodes, the braking box, the steering box, the battery management system, the OSP and the CVC. A vehicle designed for Level 3 automation would require the CVC to support efficient redundancy — especially if it includes the PCC within it, since the PCC is so fundamental to the vehicle's operation.

An external drive directly attached to the CVC can provide a central vehicle storage (CVS) unit for all of the data for the vehicle. The CVS unit would hold a complete inventory of the software, language files and other data-intensive files, such as map databases for the infotainment module or high-definition maps for automated driving.

The CVC — with expansive storage in the CVS, solid CPUs and a significant amount of RAM — would have enough resources to offer data continuity from the vehicle to the cloud. That is, it could replicate part of the cloud infrastructure at the edge to ensure that data flows easily from the vehicle to cloud servers, allowing the use of modern IT database systems and microservice architectures. Aptiv Connected Services' Connect Edge client, for example, can run analytics in the CVC to aggregate data stored in the CVS on the health of the vehicle, process it, and upload it to the cloud.

Key to making that cloud connection work well is designing the CVC to perform under harsh conditions in the field, knowing that connectivity could be unstable or uneven, depending on the vehicle's location. Developers must filter, select and optimize data for transmission over wireless networks to make the most of potentially scarce bandwidth and minimize latency — while maintaining high standards for security.

When configuring a vehicle, the OEM could create its *digital twin* — a virtual representation that reflects all of the vehicle's specifications, including all of its software capabilities. The digital twin could determine what software content was loaded onto the CVS and plugged into the CVC on the day of production. The CVC would then use that software to program the other units with the latest software before the vehicle rolled off the production line.

This storage unit would likely be a Non-Volatile Memory Express drive attached to the CVC via PCIe, which offers a good deal of flexibility when a vehicle is in the field. If a drive runs out of space, or if it degrades over time, it can be replaced.

Next steps

OEMs have already started to merge body controllers with the gateway function, but it makes sense to take this up-integration further to consolidate real-time functions, such as the PCC and data network router, into one box. The trick is to consolidate those functions in a logical and measured way while optimizing them in the context of the overall vehicle architecture and abstracting signals into services.

As OEMs build the brain and nervous system that Aptiv's Smart Vehicle Architecture™ comprises, they should not underestimate the crucial role of the CVC — the little brain that not only enables higher functions of software-defined vehicles but also quietly keeps everything moving smoothly.

ABOUT THE AUTHORS

Sylvain Pirali
Global Chief Engineer, Advanced Connectivity & Security

Sylvain Pirali leads Aptiv's product development efforts in connectivity and security, with innovations that enable OEMs to rapidly shift toward centralization of compute and manage growing software complexity. Sylvain started his career in the automotive industry in 2003 as a hardware development engineer designing body controllers. Since then, he has held various roles in product, technology and team management, giving him a broad understanding of market trends.

Aurélien Hars
Engineering Team Manager, Advanced Connectivity & Security

Aurélien Hars drives software strategy and development in advanced engineering for Aptiv's gateways and zone controllers, as well as the Automotive Ethernet backbone connecting them. Aurelien's background is in the semiconductor industry, where he participated in the MIPI Alliance's definition of UniPro, an application-agnostic transport layer for interprocessor communications. He is currently a member of the Technical Working Group in the eSync Alliance and holds patents in data transmission and automotive security.

LEARN MORE AT APTIV.COM/CONNECTIVITY →