



Centralizing Compute: How Far Can It Go?

As consumers demand an increasing number of intelligent features in today's vehicles, the approach of creating a dedicated electronic control unit for each feature is clearly unsustainable. It results in too much complexity and cost, uses too much power, adds too much weight and, most importantly, takes up too much physical space.

As a result, OEMs are up-integrating software-defined features into centralized compute wherever possible — which begs the question: How far could up-integration go, realistically? If we extrapolate this trend, will we ever see a vehicle with all compute functions handled within a single box?

While up-integration will continue, there are at least six key considerations that limit consolidation after a certain point. Vehicle designers must optimize their architectures so that they take these issues into account while maximizing the benefits of reducing the number of boxes required.



THE DESIGN FACTORS

The optimal amount of up-integration will be different for every vehicle model. OEMs have different goals in terms of active safety capabilities, comfort and convenience functions, overall vehicle design and consumer price targets. As a result, every vehicle’s electrical/electronic architecture faces unique challenges related to packaging, power management, cost and other factors. Put another way, “optimization” will mean something different for every OEM and every platform.

Across all capability levels, however — from base models to premium options — designers must weigh certain common criteria when it comes to up-integrating functions into centralized compute.

Timelines

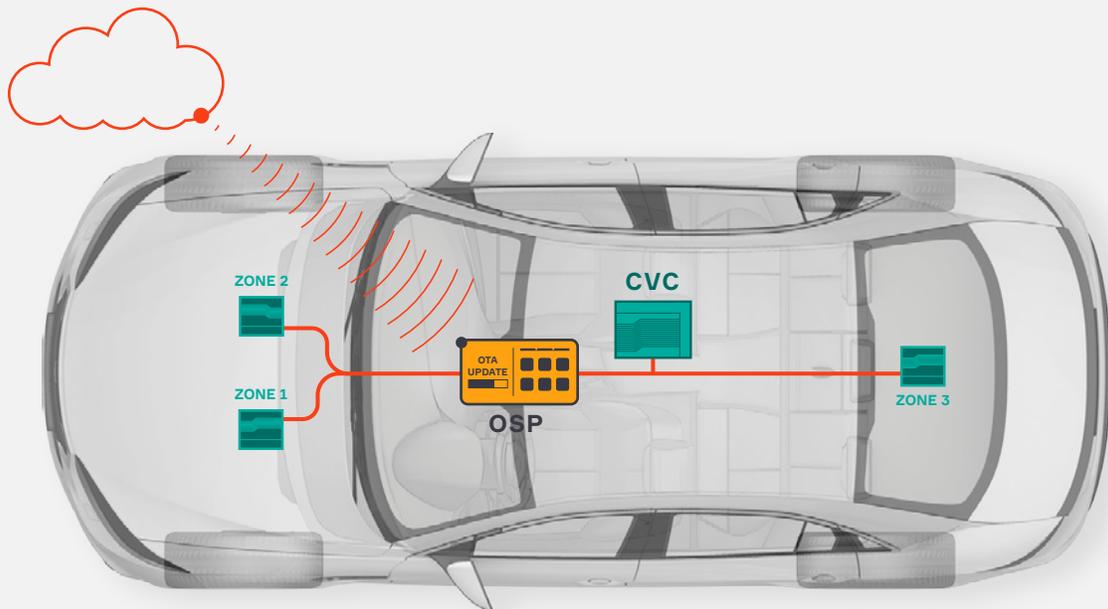
The first consideration is the update timelines of different functions, which could be on very different schedules.

For example, an OEM might want to update a vehicle’s user experience software or advanced driver-assistance system (ADAS) on a regular basis. Developers could create new applications that provide a different experience for the consumer or improve existing software to make better driving decisions. They could push those updates to vehicles over the air as often as needed.

In contrast, there are plenty of software-defined functions that are unlikely to change much over five years or more. A prime example is the power

Logically and Physically Separate

Software in the open server platform (OSP) and the central vehicle controller (CVC) can both be updated over-the-air. The difference is that the higher-level software in the OSP evolves more rapidly, while the software in the CVC handles functions that have solidified over time and are unlikely to change frequently going forward.



and body controller, which manages interior and exterior lights, window controls and door locks, climate controls, warning lights and overall power distribution.

In Aptiv's Smart Vehicle Architecture™ approach, those slowly changing or unchanging functions are handled by the central vehicle controller, or CVC. More frequently updated software resides in the open server platform, or OSP. This separation also allows for the possibility of upgrading the OSP hardware itself every couple of years as more-capable processors become available, providing a consumer experience closer to that offered by today's smartphones.

In addition, the CVC is designed to boot up very quickly and use basic signaling to get essential systems running as soon as possible — opening the door, cranking the engine and activating the rear camera, for example — while the OSP can take longer to fully boot more-complex code and use Automotive Ethernet to communicate with other systems.

Scalability

The second consideration is how much an OEM is looking to scale a particular vehicle platform. That is, if the OEM is planning to leverage the same basic platform for all vehicles, from entry level to premium, then it might opt to simply add ECUs as it increases functionality to create more premium models.

Similarly, an OEM might need to support legacy architectures for a time, and that design decision would affect the timing of a decision to up-integrate components.

Criticality

The third consideration is the criticality of the function being supported. Different systems have different functional safety requirements, and combining those systems onto the same hardware can lead to inefficiencies.

For example, the electronic stability program (ESP) controls antilock brakes, traction control and other functions that help keep a vehicle on the road and moving safely. Thus, ESP is safety-critical, and the ESP controller must be validated as an ASIL-D component for functional safety. If the ESP were to be up-integrated onto a box that also controls, say, the vehicle's audio system, then every small update to the audio would require that the entire box be re-validated at the rigorous level needed for an ASIL-D system. The same logic applies to any very safety-critical system, such as an airbag controller.

In the future, virtualization technologies could logically separate a system on a chip (SoC) into virtual machines, giving different functions their own protected spaces to prevent them from interfering with one another. Software containers could allow applications or pieces of applications to be updated independently.

Aptiv is developing middleware to support these technologies and give designers greater flexibility in where their software resides.

Redundancy

The fourth consideration is the need for redundancy. As automated driving features become more common, there are more safety-critical systems in a vehicle, because more systems are directly involved in determining where and how fast the vehicle travels. Each component should have a backup that can take over in case it fails.

This does not mean that there should be two of each component in a vehicle. That is, the vehicle does not need to have two zone controllers in the same zone, or two CVCs or two OSPs. However, it does mean installing a few extra connections.

For example, a front-left zone controller could support at least one sensor in the front-right zone, and vice versa. That way, if one of those zone controllers were to fail, the vehicle would not be completely blind in that corner.

In a similar way, the CVC and OSP can back each other up. If an OSP were to have an issue, for example, the CVC could turn off nonessential functions, such as audio and comfort features, and use its processor to handle the OSP's other functions just long enough to bring the vehicle to a safe situation.

Emerging technologies are helping to facilitate this approach. SoCs are starting to include a "safety island," which is an ASIL-D-capable section on a largely ASIL-B-capable chip. The configuration allows designers to distribute algorithms onto two ASIL-B SoCs and then use a supervisor algorithm on the ASIL-D portion to cross-check and monitor the results from both of them. If one of them shows signs of failure, the ASIL-D portion can ensure a safe switchover of critical functions to the other SoC. It will take a few years for designs to incorporate this technology widely, but there should be opportunities for careful up-integration in the future.

Space

The fifth consideration is space. While consolidating functions into fewer boxes will save space in some instances, it can create headaches in others.

For example, the vehicle doors on a premium model can contain up to 30 functions, between the automatic door locks, windows, rearview mirrors, sensors and more. If the compute that governed each of those devices were truly centralized, that would mean routing all of the wires for those devices — communication wires as well as power and ground wires — through the door grommet and into the main chassis to the central compute. That volume of wiring simply does not make sense; it adds cost while increasing the risk of wires breaking.

Instead, each door should have a door controller that can supply power to the sensors and actuators in the door while also handling communication to each device. That approach requires only five wires from the main body to the

door controller — for power and ground, and for signal transmit and receive, with a separate wire for the airbag control module.

Another example is the seat controller. Many vehicles will have one or two seats with 12- or 14-way position controls, as well as seat heating, ventilation and massage functions. Again, routing a thick bundle of wires to a seat that can move is impractical. It is better to have a seat controller mounted inside the seat, with a minimal number of wires feeding it.

A third example would be the roof. Space within an A pillar can be extremely limited, especially if it already contains airbags, so it can be tricky to route multiple wires to the roof to manage a sunroof, reading lamps and other devices.

Processing power and heat

A sixth design consideration is the characteristics of individual processors. One reason it is difficult to consolidate all computing power in a vehicle onto a single chip is that today's processors are simply not powerful enough to run an entire vehicle that way. And OEM requirements in ADAS alone are increasing at a rapid rate — too fast for silicon development to catch up.

Plus, the most powerful silicon creates a massive amount of heat. Vehicles must run in harsh environments, including high ambient temperatures, which means that getting rid of that heat can be a substantial engineering challenge. Liquid cooling is expensive and requires space for hoses. Using fans for active air cooling can generate noise, and they might break down over time.

It is much easier to use multiple processors, when possible, given that each generates less heat than a more powerful processor, and to develop simple, passive cooling solutions for them, such as heat sinks.

POSSIBLE CONFIGURATIONS

With these considerations in mind, we can get a sense of the optimal number of boxes, ranging from 12 to 19:

- 2-4 door controllers
- 2 seat controllers
- 1 roof controller
- 1 ESP module
- 1 airbag controller
- 1 powertrain and chassis controller (PCC)
- 1 CVC
- 1-2 OSPs or ADAS controllers
- 2-6 zone controllers

The number of zone controllers will vary depending on how many sensors and actuators are in the vehicle. A base model might have only two zone controllers — one for the front and one for the rear — while a fully autonomous vehicle might have six zone controllers to appropriately support all of the radars, cameras and lidars on the vehicle. Plus, electric vehicles might require additional, separate ECUs for high-voltage components, such as DC-DC converters, inverters, onboard chargers and

battery management systems. However, most configurations in the near term will include three zone controllers — two in the front and one in the rear.

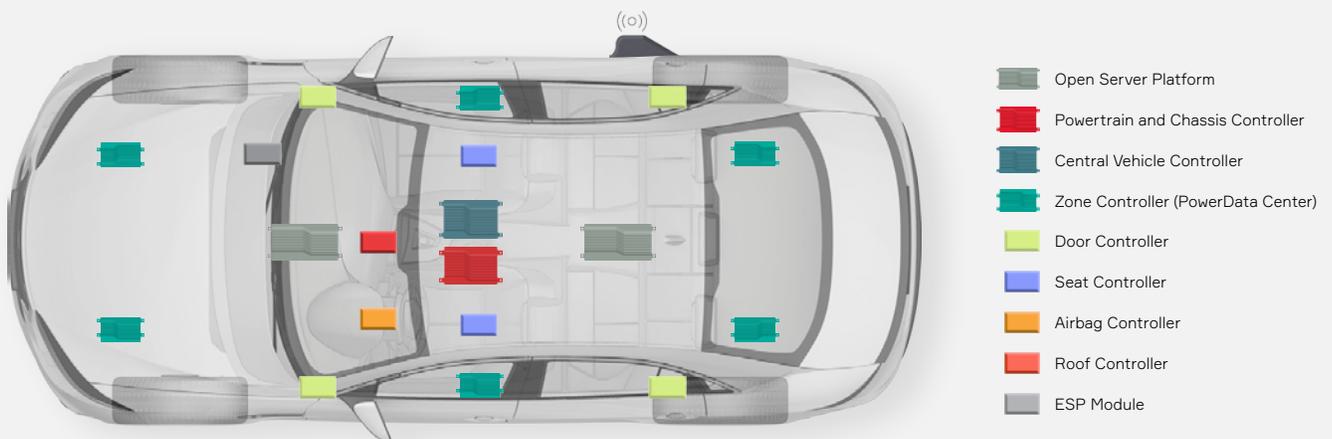
There are opportunities to consolidate further. Entry-level vehicles could reduce costs by combining CVC functions with user experience and ADAS into a single device, but this would only support up to Level 2 automated driving because it is potentially a single point of failure and requires a human driver to be the backup. Alternatively, the PCC could be up-integrated into a CVC as well.

While there are limiting factors, having 18 or fewer ECUs in a vehicle is a far cry from the 100 or more that populate some of today’s vehicles. Up-integrating functions into domain controllers and embracing a zonal architecture will serve OEMs well as they take the first steps toward next-generation architectures such as Aptiv’s Smart Vehicle Architecture™ approach. OEMs can leverage Aptiv’s unique expertise with the brain and nervous system of the vehicle to ensure that they achieve the optimal design for their vehicles’ unique needs.

Building Blocks for Success

Tomorrow’s vehicles will centralize compute where possible while maintaining the safety and robustness they will need to support an array of advanced driving capabilities as well as superior comfort and convenience features.

A highly automated vehicle could include these controllers:



ABOUT THE AUTHORS



Christian Schaefer

Vice President, Advanced Vehicle Architecture

Christian Schaefer leverages his technical expertise to advance Aptiv's Smart Vehicle Architecture™ roadmap and efficiently enable feature-rich, highly automated vehicles. During his more than 22 years of vehicle architecture experience, Christian has held roles of increasing responsibility in developing advanced electrical architecture solutions – including network communications for vehicle control, body and security, infotainment, active safety and connectivity – as well as developing cable materials and other electronics hardware. Christian believes a cross-disciplinary development approach can open possibilities to further optimize hardware, to automate production at the supplier and the OEM, and to form the basis of the software-defined vehicle – enabling revolutionary development processes and a completely different experience for consumers.



Rainer Denkemann

Director, Vehicle E/E Architecture

Rainer Denkemann is convinced that vehicle architectures are the driving factor for new innovations in the automotive industry. With more than 20 years of experience in defining new vehicle architectures, he believes in interdisciplinary teams that can optimize architectures on a system level rather than a component level. During his career he helped develop new concepts and technologies to support the evolution from early vehicle architectures to domain-centric approaches to zonal architectures – such as Aptiv's Smart Vehicle Architecture™ – which enable software-defined vehicles.

LEARN MORE AT [APTIV.COM/SVA](https://aptiv.com/sva) →