Computing Technology in Autonomous Vehicle

Fen Chen and Dong Zhao

Abstract The future of driving is quickly evolving toward AI-enabled, fully autonomous vehicles. The centralized Compute system will serve as a nerve center for all autonomous vehicles to meet stringent intelligence, performance, safety, security, and reliability requirements. We're seeing the complexity of autonomous driving systems growing at an unprecedented rate, and computational processing needs to keep pace with this growth. A high-performance, automotive-grade Compute system must be able to accommodate numerous sensor inputs from cameras, radars, light detection and ranging radars (LiDAR), ultrasonic sensors, inertial sensor module (ISM), acoustic sensors, and Vehicle-to-Vehicle (V2V)/Vehicle-to-Everything (V2X) communications concurrently to accurately and reliably perceive the environment around the vehicles. Also, it must be able to promptly enable better and safer driving decisions including prediction, planning, and control after analyzing all the perceived information. In this chapter, motivations, as well as various, Compute architectures and key components consisting of an advanced autonomous vehicle Compute system such as System on Chip (SoC), memory, storage, and network are reviewed. Furthermore, real-time operating system, onboard management, fault detection and diagnostics, security, and middleware will be illustrated. How to conduct rigid electrical tests and reliability validation to qualify autonomous vehicle Compute will be covered. Finally, challenges in Compute design, manufacturing, and validation including performance, power consumption, thermal management, size, cost, safety, security, quality, and reliability are explored for safe deployment of the autonomous vehicle at scale.

F. Chen (✉)
Cruise LLC, San Francisco, CA, USA
e-mail: fen.chen@getcruise.com

D. Zhao
Nio, San Jose, CA, USA
e-mail: Andy.zhao@nio.io

Y. Li and H. Shi (eds.), *Advanced Driver Assistance Systems and Autonomous Vehicles*,
https://doi.org/10.1007/978-981-19-5053-7_3

1 Introduction

The future of driving is quickly evolving toward AI-enabled, fully autonomous vehicles (AV). Autonomous driving (AD) is another great paradigm shift in the 100-year history of the automobile industry, which will redefine the rules of the automotive industry. The product definition of a vehicle will no longer be a "walking precision instrument" or a "computer on the wheels", but a "living space on the wheels". The role of the car OEMs will transform from a traditional car manufacturer to a Transportation as a service (TaaS) provider. Autonomous driving is an inevitable trend in the development of the industry. It is about time and life and is a key technology to reshape the future society. Since the second half of 2018, there has been a massive influx of capital into the global autonomous driving industry and the springing up of extensive new companies dedicated to the making of autonomous technologies. The prelude to the commercialization of AD has begun. The benefits of adopting AD are.

- Reduce transportation cost
- Reduce carbon emission
- Reduce riskily and distracted driving so to improve road safety
- Alleviate road congestion through higher throughput
- Offer accessibility, convenience, and independence for special needed people
- Improve human productivity and/or allow greater time for rest.

Maintaining consistent autonomous driving operations in all situations is challenging. The corner or unanticipated scenarios like the sudden onset of inclement weather or unsafe road conditions require vehicles to adapt in real-time. In general, such unanticipated cases are not the scenarios that you can code for. Only an onboard centralized Compute system that is capable of dynamically interpreting and quickly reacting can mitigate this kind of unusual scenario safely on time. Such a centralized Compute system requires the data and the ability to process that data in real-time using a combination of computing power and efficient deep learning neural networks. Therefore, the centralized Compute system will serve as a nerve center for all autonomous vehicles to meet stringent intelligence, performance, safety, security, and reliability requirements. A high- performance, automotive-grade Compute system houses the central Compute and connectivity to accommodate vision, radar, ultrasonic radar, acoustic sensors, ISM, and LiDAR signal transmissions. It must be able to accommodate massive data from numerous sensor inputs and V2X communications concurrently to accurately and reliably perceive the environment around the vehicles. Figure 1 illustrates how an AV's Compute sees and detects surrounding objects such as vehicles, pedestrians, and traffic lights by sensors on a rainy day. AD requires a much greater and more reliable awareness of everything around the vehicle as compared to traditional vehicles. The AV Compute is required to understand what they are "seeing" and the ability to control the vehicle to adapt to the situation evolving outside the car. It should be noted that this requirement is dramatically different from the Compute required by simpler Advanced driver assistance system (ADAS) functions like adaptive cruise control or emergency braking.

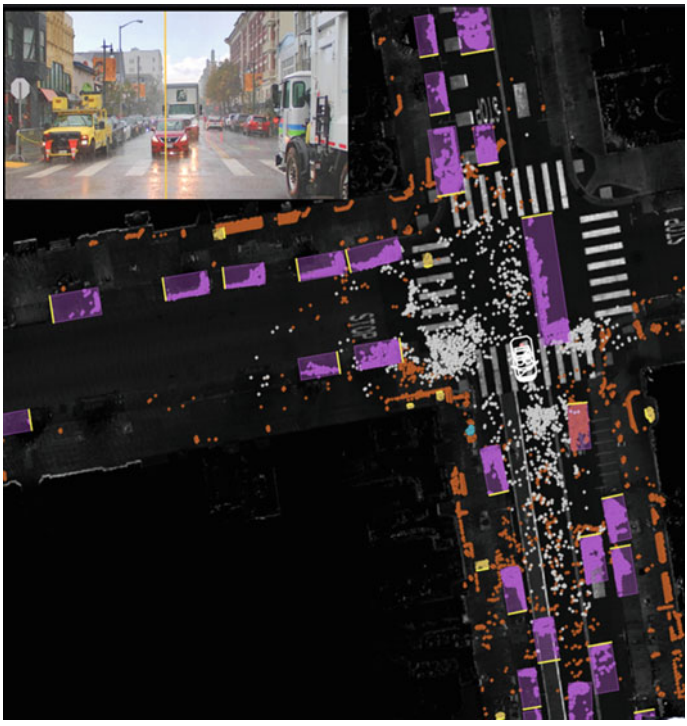

Fig. 1 Onboard Compute perception of surrounding environment around the AV

Maintaining consistent ADAS and autonomous driving (AD) operations in all situations requires machine learning, computer vision, and sensor fusion. Machine learning, computer vision, and sensor fusion will play critical roles in next-generation AVs. The high-performance AV Compute must be able to enable better and safer driving decisions including prediction and planning and must control promptly after analyzing all the perceived information as demonstrated in Fig. 2. To increase overall neural network capacity and boost the performance and responsiveness of automated driving perception systems, a Compute with high-speed parallel processing and massive processing acceleration becomes a key requirement.

One of the key missions of AV technology is to improve road safety to reduce the road fatality rate. World Health Organization (WHO) reported that about 1.27 million people die due to road traffic accidents each year [1]. Safety is a critical part of the AV Compute system. With AV, we are essentially to use a sophisticated Compute system to replace the human driver to make a safe decision. International standard ISO 26262 was developed for traditional automotive electric/electronic systems. ISO26262 defines functional safety features and requirements for all automotive electronic and electrical safety-related systems [2]. However, AV is not within its scope. For AV, there is a need to implement more rigorous safety standards and certifications to assure the highest levels of passenger and environmental safety. A

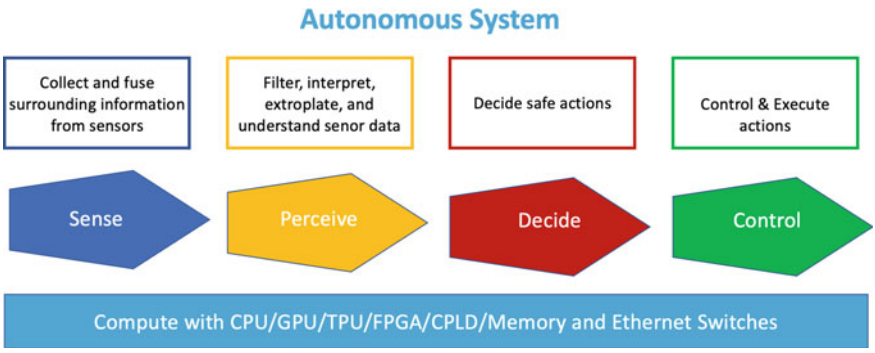

Fig. 2 An illustration of an autonomous system for AV

high-performance AV Compute must be built from the ground up to meet desired safety requirements, from development to validation to deployment. It is essential to address safety needs during the early stage of design cycles. A design for safety must be embedded in the design from day one to guarantee true automotive-grade safety conformance.

Reliability is another critical part of the AV Compute system, and it is closely tied to vehicle safety. Without a human driver, any hardware performance degradation and failures including soft failures due to performance regression and an intermittent malfunction could trigger a fatal accident. A Compute system must be able to be functional with top performance and must respond faster than the human driver at all times to guarantee AD safety. It is well known that hardware failure rates could be high during the early vehicle usage life and late wear-out period. As there have been no specific safety rules and inadequate field safety lessons learned for AV, improving an overall reliability target to reduce Compute failure rates of both hard and soft failures is essential to guard band the safety of AV. On the other hand, AV hardware, in general, will have its unique mission profiles. As shown in Fig. 3, the trend of vehicle operating time increases from traditional vehicles to Robo-taxi AVs. AV has 2–3 times of life mileages and vehicle operating hours as compared to traditional vehicles. In general, if AVs are used for road-sharing taxi business, vehicles will be required to be operative for over 20 h per day to maximize their business profit goal. With such longer daily continuous operation hours or mileages, the reliability specifications for AV hardware especially for Compute are high and challenging. A superior reliability resilience, which ensures the continuity of reliability and safety throughout the entire AV life cycle, is required for AV Compute.

This chapter presents state-of-the-art Compute systems for AD, covering five key performance metrics and nine key hardware enablement technologies, followed by validation and challenges to realize AV Compute operational performance. The remainder of this chapter is organized as follows. Section 3.2 discusses the general architectures of computing systems for AD. In Sect. 3.3, we show six key hardware and three key software constituents of an advanced Compute system. In

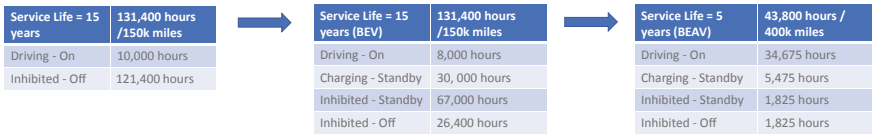

Fig. 3 Mission profile comparison among traditional vehicles, battery electric vehicles, and battery electric autonomous vehicles

Sect. 3.4, functional tests and validation of the AV computing system are introduced. Section 3.5 presents possible challenges for large-scale deployment. Finally, this chapter concludes in Sect. 3.6.

2 Compute and ADAS Technology

AV utilizes high-performance computing platforms together with a complex software system to enable a real-time AI-based perception and decision-making for vehicle maneuvers. A sophisticated AD algorithm in general requires a high volume of sensor data and a complex computational pipeline. Therefore, a Compute needs to process an enormous amount of data in real-time with extremely small latency. As the level of autonomy increases, the data generated by the AV will become larger and larger. According to Intel's estimation [3], assuming that an AV is equipped with GPS, ultrasonic sensors, camera, radar, and LiDAR sensors, the data generated by the above-mentioned sensors per second is shown in Table 1. For a vehicle driving about 20 h per day, AV Compute needs to process about 8 TB of data every day.

How to enable AV Compute to process such a large amount of data in real-time, and then based on the extracted information, make logical decisions that control safe driving behaviors is a challenge.

Two key questions to solve the above problems are:

1. Where the data processing is done: distributed-based architecture, centralized-based architecture with one central processing unit, or a hybrid-based architecture with several decentralized computing units?
2. How to transmit data from the sensors to the central processing unit: when data fusion is performed on multiple sensors that are not located in one place but spread

Table 1 Typical data size generated by various sensors per second for AV

GPS	50 kB
Ultrasonic radar	10–100 kB
Camera	20–40 MB
Radar	10–100 kB
LiDAR	10–70 MB

over the different locations of AV, the connectors and wiring cables between the sensors and the central processing unit need to be specially designed.

As shown in Fig. 4, in a complex network, three typical network structures: Centralized, Decentralized, and Distributed, are referenced.

Centralized Computing Architecture The central Compute will receive and process the raw data transmitted by each sensor. The central Compute will make and execute the decision. With the fusion of sensor data, each sensor knows what each other sensor is doing.

Decentralized Computing Architecture is a mixture of Centralized and Distributed computing solutions. Several preprocessing Computes process various sensor data before sending them to the high-level central Compute.

Distributed Computing Architecture each sensor processes its data to a certain extent, and also makes decisions locally. Only object data is transmitted from the sensor to the central Compute. The central Compute integrates the object data from each sensor first and then makes decisions and executions.

There are pros and cons for each architecture [4]. For distributed computing architecture, the advantages are that each sensor terminal processor does not have to process a large amount of data at once, and there is less demand on how to transmit data from the sensor to the central Compute safely and efficiently. A lower bandwidth, simpler and cheaper interface can be used between the terminal sensor and the central Compute. In most cases, a bandwidth of less than 1 MB per second is sufficient. Since a lot of data processing is done at the terminal processor, the increase in the number of sensors will not greatly decrease the performance of the central Compute. Since the central Compute only needs to integrate the object data, it has lower requirements on computing power and lower power consumption. It can combine various sensors in a cost-efficient way. Its disadvantages are that this computing architecture must distribute information at the same time and synchronize the information among all sensor nodes. When the number of nodes exceeds 3–4, this approach has almost become very difficult. The central Compute obtains object data rather than actual sensor data, so it cannot real-time track a specific "areas of interest" event. Also, as the terminal sensor needs to be equipped with a processor, its volume will be larger,

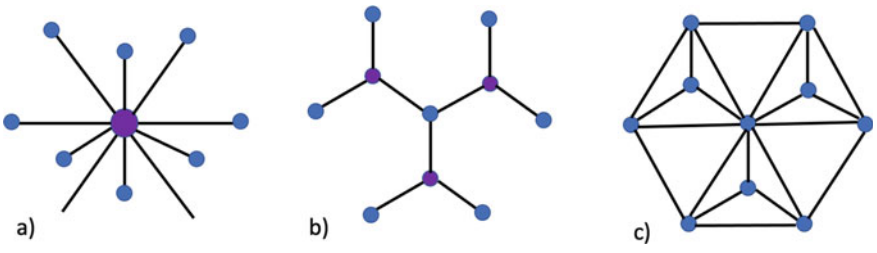

Fig. 4 Three typical network structures: **a** Centralized, **b** decentralized, **c** distributed

and the overall price and power consumption will also be higher. Since sensors need to process data and make decisions locally, their requirements for functional safety will also be higher.

The advantages of centralized computing architecture are that the cost and power consumption of terminal sensors are low since it only needs to complete the task of sensing and transmitting data. Therefore, the requirements for functional safety are low for terminal sensors. The sensor size can be small, so the installation space required is small. The installation position is also flexible and the replacement cost can be low. On the other hand, the central Compute will get the best quality information. The reason is that if the terminal sensor does not modify the data or filter the data, the central Compute can obtain the maximum possible information or original data to make the correct decision if needed. The disadvantages are that the central Compute will become a "big monster". GB-level data must be transmitted from various sensors with ultra-low latency and such massive data must be processed in time by the Compute without any delay. Broadband communication of up to several GB per second is required for data transmission and collection in real-time, which may result in high electromagnetic interferences. The central Compute needs powerful computing power and speed to process all the data transmitted from the terminal sensors, which consumes a lot of power and generates a lot of heat. In addition, the increase in the number of terminal sensors will greatly decrease the performance of the central Compute. If the central Compute is non-scalable, it will not be able to provide the required functional performance for AV needs with AV technology scaling up.

Terminal sensors are always needed to process data locally which can reduce bandwidth requirements and help to reduce AV costs. On the other hand, a centralized Compute is always needed to integrate the information of all terminal sensors to complete the overall perception of the vehicle's surrounding environment and make decisions for AV pathfinding, maneuvering, and motion trajectory. Therefore, the hybrid decentralized computing architecture that finds the optimal combination of distributed and centralized architectures is more likely to be the final technology path.

As currently, most AV Compute prototypes are centralized, we will use it as our AV computing system reference architecture. Generally, per functionality, the AV Compute can be divided into computation, network and communication, storage, and power supply management. The following sections will discuss the corresponding components in more detail.

2.1 Levels of Autonomous Driving

SAE International (Society of Automotive Engineers International) published the revised version of the autonomous vehicle classification standard in 2018. It defines six different levels of automation, ranging from Level 0 (no automation) to Level 5 (full automation), known as SAE J3016. Currently, as shown in Fig. 5, most vehicles

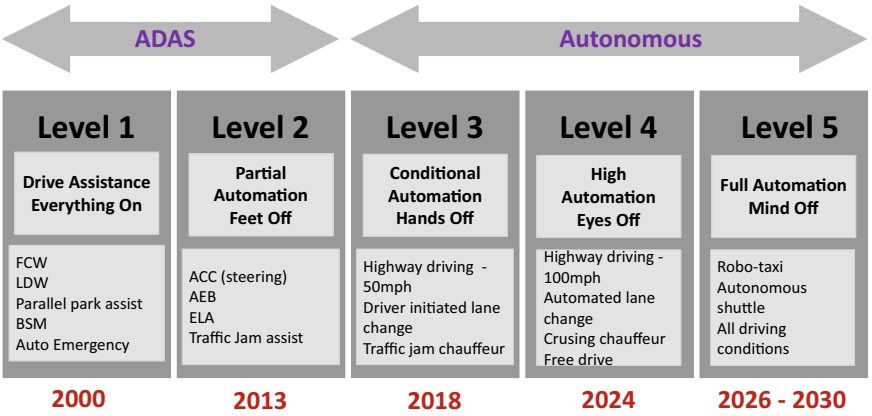

Fig. 5 Progress from Level 1 to Level 5 autonomous vehicle with timelines

on the road are only at SAE levels 0 to 2 with ADAS functionality. For such levels, human drivers remain the key controller of the vehicle to make driving decisions and are responsible for all potential hazards that occurred during driving. Every advancement in automation level requires substantial hardware and software technology advancements, and proper management of all safety–critical functions.

2.2 Platform for Autonomous Driving System

AV has two meanings: "intelligence" and "ability". The so-called "intelligence" refers to the ability of the vehicle to perceive, synthesize, judge, reason, decide and remember as intelligently as a human. The so-called "ability" means that the AV can ensure the effective execution of the "intelligence", implement active control, and be able to perform human–computer interaction and collaboration. Autonomous driving is an organic combination of "intelligence" and "ability". The two complement each other and are indispensable.

To realize "intelligence" and "ability", the core competencies of an autonomous vehicle system can be broadly categorized into four categories: environment perception, localization, decision-making and planning, and vehicle control. Similar to the human driver's perception of the driving environment and vehicle status through visual, auditory, and tactile sensory systems during driving, the AD system acquires its status and surrounding environment information by configuring internal and external sensors. Internal sensors mainly include vehicle speed sensors, acceleration sensors, wheel speed sensors, and yaw rate sensors. Mainstream external sensors include cameras, lidars, millimeter-wave radars, ultrasonic radars, and positioning systems. These sensors can provide massive amounts of information about the driving environment in all directions. To effectively use this kind of sensor information, it is

necessary to use sensor fusion technology to combine the independent information, complementary information, and redundant information of a variety of sensors in space and time according to certain criteria, to provide an accurate understanding of the surrounding environment and own motion status. The decision-making planning subsystem represents the cognitive layer of autonomous driving technology, including two aspects of decision-making and planning, rule-based and AI-based. Figures 6 and 7 illustrate the AV process control flow and configuration schematic for AV software and hardware, respectively.

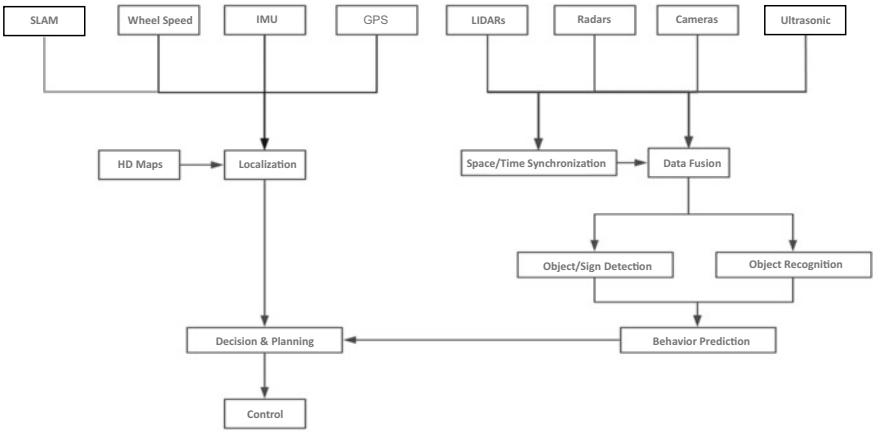

Fig. 6 The flow chart of an autonomous driving software system

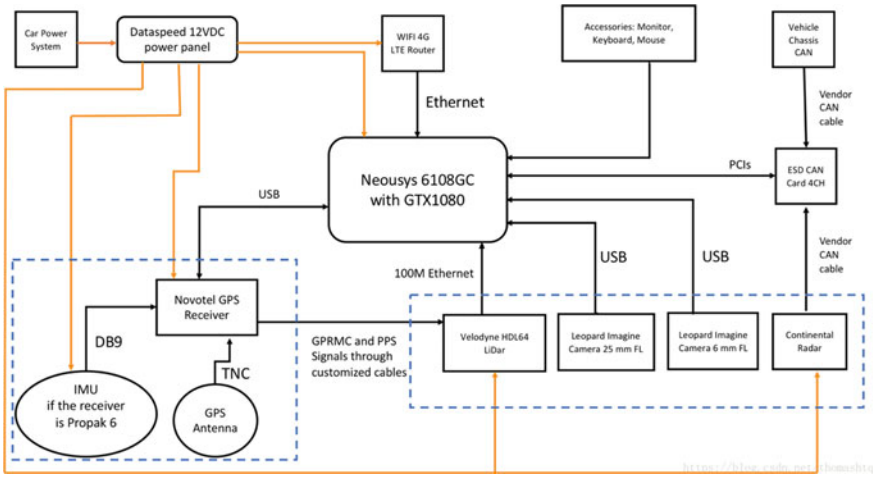

Fig. 7 The schematics of a hardware configuration for the Baidu Apollo AD system [5]

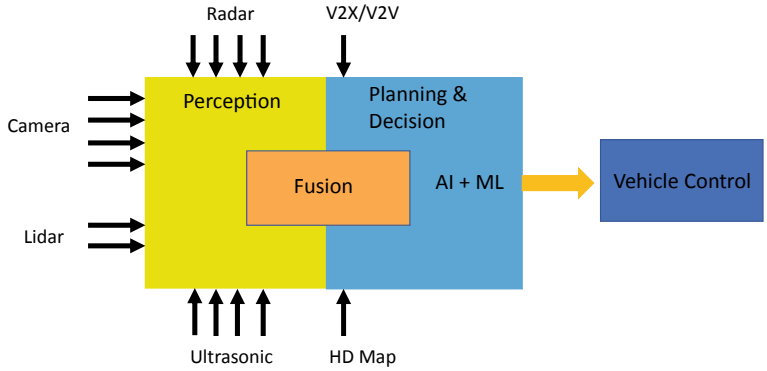

Fig. 8 AV with V2X and V2V for perception, planning, decision, and control

The decision-making system defines the interrelationship and function allocation between the various parts and determines the safe driving mode of the vehicle. The planning part is used to generate a safe, real-time collision-free trajectory. The vehicle control subsystem is used to realize the vehicle's longitudinal distance, vehicle speed control, lateral vehicle position control, etc. It is the final executive mechanism of vehicle intelligence. Environmental perception and decision planning correspond to the "intelligence" of the autonomous driving system, while vehicle control reflects its "ability".

To realize L4 or L5 autonomous driving, it may not be enough to rely just on the "smartness" of a single AV. As shown in Fig. 8, Vehicle-to-Vehicle (V2V) and Vehicle-to-everything (V2X) communications can be leveraged to achieve further improvements in areas of perception and/or planning through vehicle cooperation. Road conditions and traffic data through the V2X and V2V can provide more information than the internal and external sensors of a single AV. Such real-time data together with the help of a high-definition 3D dynamic map can enhance the perception of the environment for the no-line-of-sight situations. For example, under severe weather conditions such as rain, snow, and heavy fog, or in challenging scenes such as intersections and corners, radar and cameras cannot clearly distinguish the obstacles ahead. V2X and V2V can be used instead, which can realize an intelligent prediction of road conditions and avoid accidents.

2.3 Perception and Localization

Perception and localization are two of the most critical parts of AV Compute. Without the quantitative perception of the 3D environment around the vehicle, the decision-making initialized by Compute cannot work properly. Perception refers to the ability of an AV to collect information and extract relevant knowledge from the environment. AV needs to develop a capability to understand the surrounding environment such

as road obstacles, road signs, and the movements of other road agents. Localization refers to the ability of the AV to determine its position concerning the environment. The main tasks of perception and localization include vehicle position, motion status, object detection, and object tracking.

Environment perception tasks can be fulfilled by using radars, ultrasonic sensors, LiDARs, cameras, and IR cameras, or a fusion among them to extract road traffic conditions and on-road object detection. Different sensors have different strengths and weaknesses. Ultrasonic radar is mainly used for vehicle reversing due to its limited reaction speed and resolution. Millimeter-wave radar and LiDAR are responsible for the medium and long-range environmental perception. LiDAR can produce 3D measurements and detect object traveling speed. But it offers little information on objects' appearance. The camera is mainly used for the identification of traffic lights and to provide rich appearance data with much more details on the objects. But its performance is not consistent especially under dark illumination conditions. It also does not implicitly provide 3D information. Therefore, sensor fusion is required to make full use of the advantages of each sensor.

Localization is the task to determine the pose of the ego vehicle (position and orientation) and measuring its motion. Knowledge of the ego vehicle's position is a critical piece of information that enables AV Compute to execute safety-related, AD maneuvers. One of the most popular ways of localizing a vehicle is the combination of satellite-based navigation systems, inertial measurement units (IMU), and a high-definition HD digital map. Satellite navigation systems, such as GPS and GLONASS, can provide a consistent outcome on the global position of the vehicle. However, the use of GPS and GLONASS requires reliable service signals from space satellites and the update rate is comparatively low. Inertial measurement units, which use an accelerometer, gyroscope, compass, and signal processing techniques to estimate the attitude of the vehicle at a very fast update rate (every 5 ms), do not require external infrastructure. However, IMU's accuracy is not great, and the error accumulates over time. In general, Kalman Filter techniques are used to combine the advantages of GPS/GLONASS and IMU to provide accurate and real-time position updates. Map aided localization algorithms use local features to achieve highly precise localization and have seen tremendous development in recent years. In particular, Simultaneous Localization and Mapping (SLAM) is a promising method, which refers to a process in which a moving object calculates its position based on the information from the sensors while constructing a real-time map of the environment. There is also a dynamic HD map-assisted localization method, which is a sensor system-based DeepMap with super dynamic perception capability. It can deliver road information (road geometry information, congestion information, construction information, etc.) and obstacle information (position, speed, type, etc.) to the AV in real-time through a high-performance AD cloud infrastructure. To implement this method, fast data collection and transmission capabilities are required.

2.4 Prediction, Planning, and Control

Environmental perception and localization mainly play a role in determining the state of the external environment and provide a basis for decision-making and planning. The task of prediction, planning, and control for Compute is to continuously provide collision-free decisions and motion trajectories from the current pose of the vehicle to the given destination, taking into account system dynamics, obstacles, and possibly desired criteria such as trip frequency, travel time, cost and conformable ride function. In the decision module, the main problem to be solved is how the vehicle should go. This is divided into two aspects, namely path planning and behavior planning.

1. Path planning

Path planning generally is a computational work to find a sequence of road paths to move the object from the source to the destination. It is a technology in the field of high-precision maps. In the traditional human-driving mode, if there is an error in the map navigation, it can be corrected by a human driver. In the field of autonomous driving, map accuracy and navigation accuracy will directly affect AV safety and user experience. Therefore, the high-precision map is very important. Path planning is the problem of finding the shortest path between two points. Commonly used algorithms for finding the shortest distance include Dijkstra, Floyd, A*, and RRT algorithms.

2. Behavior planning

The behavioral planner focuses on the AV on-road behaviors to assure it follows road rules and interacts with other agents safely. The prediction of traffic agents can be achieved through a variety of algorithms, and a set of motion models can be constructed. There is a lot of uncertainty in the behavior of other vehicles on the road such as accelerating and turning. The commonly used solution is to use Gaussian noise to represent the uncertainty of traffic participants. Because most of the participants' behavior must follow a normal distribution, the entire model construction can be regarded as a Gaussian process. The prediction of the behavior and intentions of traffic participants can be regarded as a dynamic time series process, and the corresponding problems can be solved by using convolutional neural networks (CNN).

Speaking of the vehicle itself, the local motion planning that requires decision-making includes: driving, following, turning, changing lanes, stopping, etc. How the vehicle makes decisions needs to be judged dynamically. The overall process of vehicle own motion planning should be divided into four steps as shown in Fig. 9. The first step is to perceive the changes in the environment. As an example, if a vehicle in front of the AV starts to merge into the lane that AV is currently using, per the perception of the local scene, a model should be used for prediction and decision. The final behavior output of AV maybe just slow down or change into another lane to assure safety with local goal setting. During the decision-making process, other vehicle behaviors and whether they comply with road rules and regulations must also be considered. The overall decision-making process of each behavior could be long, and each decision-making step affects the other. Therefore, the function of this

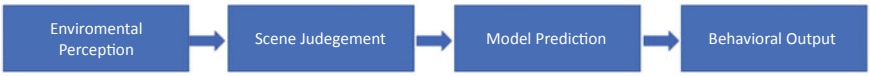

Fig. 9 Vehicle own motion behavior planning process

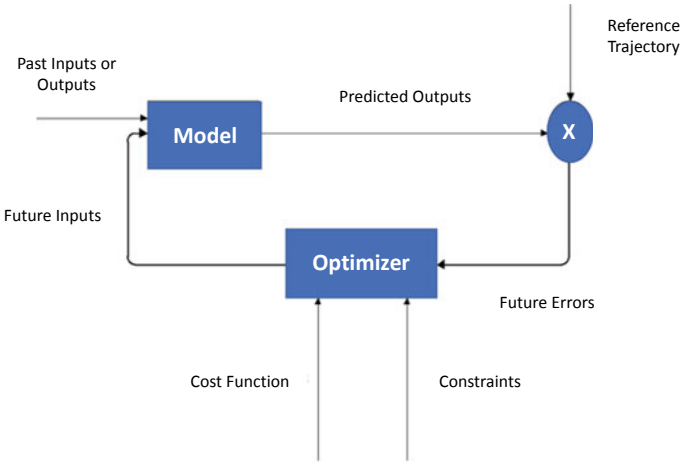

Fig. 10 Basic structure of MPC

kind of AV behavior decision-making can be regarded as a series of probabilistic additions, which can be modeled as a Markov decision process.

After environmental perception and decision-making planning, it comes the step of execution control. The execution control is a critical task for Compute. How to transmit the decision to the functional hardware components of the vehicle and implement the required accelerator, brake, steering, and shift commands are the keys to controlling the vehicle maneuvers. The most feasible solution for AV to control the behavior of each component is through the CAN power bus, and transmit instructions to each component through electronic signals. Autonomous systems need motion models for control execution. A control approach that uses system modeling is commonly referred to as Model Predictive Control (MPC). Figure 10 illustrates the basic structure of MPC. As shown in Fig. 10, it optimizes the control input by minimizing an index function while satisfying constraints and guaranteeing vehicle safe operation.

2.5 Functional Safety

As the complexity of the AV system continues to increase, new technologies will introduce new safety risks. Uber's self-driving car accidentally killed a pedestrian

in March 2018. The US National Highway Transportation Safety Administration launched an investigation into Tesla's Autopilot feature, which has allegedly been involved in 11 collisions with stopped cars, resulting in one death and 17 injuries, over the past three-and-a-half years. Volvo issued a large-scale recall notice to the global market in March 2020. The number reached 700,000 vehicles, involving nine models on sale. The reason for the recall was that Volvo had previously conducted a safety test on the XC60 in Denmark. It was found that the Autonomous Emergency Braking (AEB) system did not stop the vehicle in time in the event of a collision as expected. A fatal crash involving NIO's autopilot function happened in August 2021, and the driver was killed while driving NIO's ES8 SUV. Therefore, the safety of AVs has received a lot of attention.

Function Safety (FUSA) refers to failure behaviors caused by hardware and software failures or unexpected behaviors in the design of automobiles. It is defined as safety due to the absence of unreasonable risk and is only concerned about malfunctioning systems. Currently, automotive safety frameworks include ISO 26262, the functional safety standard, and ISO/PAR 21448.1, the safety of the intended functionality. ISO 26262 standard defines the functional safety terms and activities of electrical and electronic systems in motor vehicles. Therefore, it can only solve the hardware and software hazards that may affect the safety of autonomous vehicles. The standard defines four automotive safety integrity levels (ASIL). ASIL D is the most stringent, and A is the smallest. Each level is associated with its specific development requirements, which must be complied with during certification. ISO 21448 SOTIF pays special attention to failure causes related to system performance limitations and predictable system misuse. Either hardware technical limitations (such as sensor performance limitations and noise) or software algorithm limitations (such as target detection failures and actuator technical limitations) could result in limited performances or insufficient functions for AV operation. User misuse such as overload and confusion could result in failures of AV operation as well. SOTIF is designed for Level 0–2 autonomy. SOTIF can be viewed as an extension of the functional safety process, specifically designed to solve the challenges of autonomous driving functions. SOTIF also uses hazard analysis and risk assessment (HARA) to identify hazards due to performance limitations and abuse. To demonstrate that the safety requirements are met for AV Compute, a process of design for safety, unit testing, and system verification needs to be thoroughly conducted.

Regarding safety risk mitigation, an intelligent driving safety system should be implemented to provide safety analysis and real-time monitoring services for potential problems in the perception, decision-making, and control modules of AV. Based on the concept of expected functional safety, the driving scene and system safety are analyzed and evaluated to improve the safety of AD. The driving behavior of AV highly depends on the stability, intelligence, and safety of the AV hardware and software systems. The main sources of safety risks for AV are as follows:

1. Hardware safety

Compared with traditional cars, AVs do not require the human driver to directly control the vehicle. But instead, it transfers part or all of the vehicle control to

the automatic control system. AV vehicle motion perception and sensor data fusion functions play a decisive role in AD. Whether the hardware architecture setting is technically sound and sophisticated or not, whether the Compute and controller settings are comprehensive or not, and whether the sensors can quickly and accurately obtain road environment information or not, all of them would induce hardware safety risks.

2. Software reliability

Compared with traditional cars, the software development time for AVs is not long enough. Thus, it is lacking extensive supporting field data. The AV technology itself is still under development so it is not yet mature. The AV software system needs long-term reliability analysis. Therefore, its safety and stability still need long-term monitoring and validation.

3. Environmental security

When making driving decisions, AV still needs the correct driving of other on-road agents. Only when other agents have the correct driving behaviors, AV then will make its own correct decision and reasonable operation.

To ensure proper and safe functionality of AV, the development has to consider not only hardware but also software and user misuse at both component and vehicle levels. A holistic and traceable approach for risk analysis, risk mitigation, test specification, and validation is needed to orchestrate the behavior of single products in the function chain. For the AV Compute system, all embedded integrated circuits need to meet ASIL-C or even ASIL-D levels, and they shall be qualified by AEC-Q100 standards. AEC-Q100 is a set of stress test standards designed by AEC mainly for integrated circuit products for automotive applications. This specification is very important for improving product reliability and quality assurance. To prevent various conditions or potential failure states that may occur, AEC-Q100 conducts strict quality and reliability standard-based validation for each chip.

3 Advanced Computer System

3.1 Architecture Solution and Comparisons

The key to the success of an AV is to make a reliable decision in real-time quickly. Reasonable selection of AV Compute platforms to complete real-time large-scale sensor data processing, real-time driving prediction, and real-time control are essential to the safety, reliability, and durability of AV operation. During the early stages of AV development, most AV Compute solutions started with an Industrial Personal Computer (IPC) using the architectures based on Intel CPU + Nvidia GPU platform. IPC is a ruggedized and enhanced personal computer that can be used as an industrial controller to operate reliably in an industrial environment. The use of a

fully sealed industrial chassis that meets the Electronic Industries Alliance (EIA) standard enhances the ability to resist electromagnetic interference. The CPU and various functional modules all use a plug-in structure with a soft locking lever to improve shock and vibration responses. The overall architecture design for AV needs to consider the requirements of ISO26262. The CPU, GPU, FPGA, and bus are all designed with redundancy to prevent single points faulty failure. Even when the IPC system fails, the MCU still can serve as a final guard bander, and directly send instructions to the vehicle Can bus to execute the emergency pull over or stop of the vehicle. At present time, this centralized architecture puts all computing tasks into one industrial computer. Therefore, the Compute size is large and the power consumption is high. But this architecture is very convenient. With the traditional X86 architecture, a computing platform can be built very quickly, and the card slot design is also convenient for hardware updates. As an example, Fig. 11 shows the Baidu Apollo AV adopted Neousys Nuvo-6108GC IPC for AV application. Nuvo-6108GC is the world's first industrial-grade Edge AI Computer supporting high-end graphics cards. It's designed to fuel emerging GPU-accelerated applications, such as autonomous driving, by accommodating Intel Xeon E5-2658V3 12-core CPU and Nvidia RTX 3070 GPU [6]. The peak CPU operating speed is 400 frame/s and it requires 400 W. Each GPU is capable of 8 TOPS performance computing and it requires 300 W. The whole system consists of two independent Computes. Therefore, the whole system can provide 64.5 TOPS Compute performance, and requires 3000 W. If both Computes operate at their maximum loading, a total of 5000 W will be required and it would produce excessive heat.

While providing high-performance data processing support, the Compute platform on a vehicle also needs to take into account issues such as power consumption, heat dissipation, and switch interface, which are equally important for continued safe driving.

- Power budget

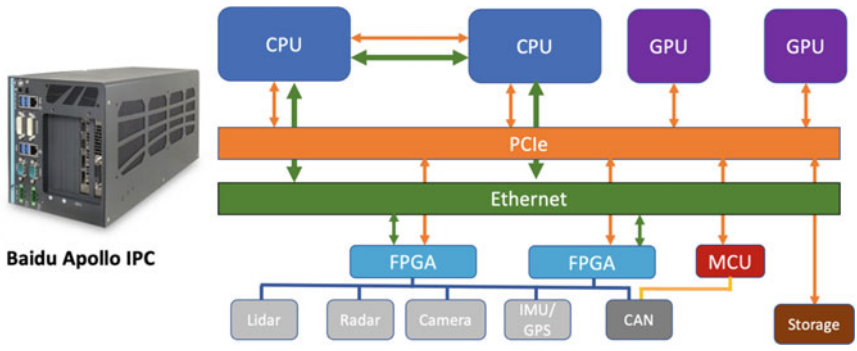

Fig. 11 Baidu Apollo IPC-based Compute platform block diagram

Vehicle platforms, especially electric vehicles, mostly provide a 12 or 48 V DC power supply. Compared with the traditional 220 V AC power supply for data center Computers, Compute used for vehicles needs to adapt to a 12 or 48 V DV power supply. For EV, the power consumption of Compute could reduce the vehicle mileage significantly. Therefore, the maximum power that an EV power supply can support is also an important issue for Compute design.

- Cooling solution

The heat dissipation solutions that can be used in vehicles are mainly air-cooled and liquid-cooled. Although the cost of air cooling is relatively low and the structure is simple, the disadvantage is that the heat dissipation efficiency is low and the noise is also relatively large. For most battery electric vehicles or hybrid vehicles, a liquid cooling loop for batteries already exists. Therefore, for battery-powered AV, the liquid cooling solution is a natural choice for a Compute mounted on a vehicle.

- Connector interface

In the existing AV Compute platform, different computing units are connected through an Ethernet Switch or PCIe Switch to transmit large amounts of data and complete coherent calculations. However, traditional ethernet mostly uses the RJ45 interface form, which will have a certain impact on the stability of the network during the long-term AV operation. In addition to the AI calculations, Compute also needs to coordinate and control various electronic control units (ECUs) and mechanical components in the vehicles to complete the driving control operations. This is achieved by the interconnection through the communication bus. Communication buses such as CAN, USB3.0, LIN, serial port, etc. are commonly used as interfaces to fulfill vehicle data sharing and effective transmission of control instructions from the Compute to vehicle ECUs and mechanical components. In general, with an increasing number of sensors on the vehicle, more interfaces are needed to connect all the required sensors.

- Real-time

AVs have very high requirements for system response in real-time. For example, in dangerous situations, the vehicle braking response time is directly related to the safety of vehicles, passengers, pedestrians, and roads. The braking reaction time includes not only the vehicle control time but the response time of the entire AD system including the time for perception, prediction, planning, and control. If the braking distance of the vehicle at a speed of 65 miles/h is to be less than 30 m, the overall response time of the system cannot exceed 100 ms, which is close to the response time of the best F1 players. Divide the response of AD into the requirements of each functional module of its Compute platform in real-time, including:

- Time for detection and precise positioning of surrounding targets: 15–20 ms.
- Time for data fusion and analysis of various sensors: 10–15 ms.
- Behavior and path planning time: 25–40 ms.

With the development of AD technology, the AD algorithm is constantly improving. After the algorithm is solidified, a dedicated ASIC chip or FPGA chip can be used to integrate the sensor and the algorithm to realize the edge calculation inside the sensor. This approach can further reduce the number of computing demands of the Compute to reduce power consumption and Compute size. Currently, there are two common schemes to construct an AV Compute platform:

1. Adopting off-the-shelf mature solutions

Some Compute platform solutions with different architecture designs for ADAS and AV applications are available. Designs are based on graphic processor unit (GPU), field-programmable gate arrays (FPTA), application-specific integrated circuits (ASIC), and digital signal processors (DSP). Among them, Nvidia's Xavier-based Drive PX2 and Drive AGX Pegasus platforms are two popular Compute solutions incorporating Nvidia's extensive experience in the field of Deep Learning for AV applications. Nvidia has been deeply involved in the field of autonomous driving for many years and is expected to become a new Tier 1 for the AV market. The main advantages of PX2 and AGX are:

- It is a complete system offering a turnkey solution. It is designed by the standards of vehicle regulations.
- Many sensors have been adapted already, especially image signal processor provided to adapt to many cameras. It can execute processes like demosaicing, noise reduction, auto exposure, autofocus, and the auto white balance at high speed and high quality.
- It has a relatively clear roadmap to facilitate subsequent iterative upgrades.

Nvidia Drive AGX is a powerful autonomous machine SoC [7]. Each Drive AGX consists of two Xavier SoCs and two Turing Tensor Core GPUs. Each Xavier has a custom 8-core Arm-based CPU. Drive AGX is capable of 320 trillion operations per second (TOPS) for Al computing and safe AV operation. The platform is designed and built for L4 and L5 autonomous systems.

GPU can provide tens to hundreds of times the CPU performance in terms of floating-point calculations and parallel calculations. Using GPUs to run machine learning models and perform localization and detection has greatly reduced the time consumed by CPUs. Relying on its powerful computing capabilities and driven by the rapid development of machine learning, GPUs are currently very popular in the deep learning chip market. Many car OEMs are also adopting GPUs as sensor data processing chips to develop AV. Therefore, GPUs have become the mainstream trend. However, the weakness of Nvidia's solution is that the performance of its CPU as a part of the SoC is still not powerful enough. Its CPU based on the Arm architecture has a main frequency of only 1.8 GHz and eight cores, which likely is difficult to meet the computing performance requirements for some AV applications.

Other commercially available solutions are Xilinx's Zynq UltraScale + ™ MPSoC ZCU104 product, TI's TDA3x, and Mobileye's EyeQ5 products as examples. Zynq is an FPGA-based SoC including 64-bit quad-core ARM Corte-A53 and dual-core ARM Cortex-R5. It is built by a 16 nm FinFET semiconductor technology node.

It is claimed to achieve 14 images/s/W for running convolutional neural network (CNN) tasks [8]. As a strong competitor of GPU in algorithm acceleration, FPGA has flexible hardware configuration, low power consumption, high-performance, and programmable advantages, which is very suitable for perceptual computing. More importantly, FPGAs are much cheaper than GPUs. In the case of energy consumption as a major concern, FPGAs have obvious performance versus energy consumption advantages over CPUs and GPUs. The low power consumption of FPGA makes it very suitable for sensor data preprocessing. In addition, the continuous development of perception algorithms means that the perception data processor needs to be constantly updated, and FPGAs have the advantage of hardware upgradability. One of the disadvantages of using FPGA is that it requires knowledge of hardware-level programming, which is difficult for many software developers. Therefore, FPGA is often considered an exclusive architecture for experts. However, some software platforms have emerged specifically for FPGA programming, which makes it possible for more software developers to use FPGAs. With the rapid popularization of the combination of FPGA and sensors, and the further optimization of vision, voice, and deep learning algorithms on FPGA, FPGA is very likely to gradually replace GPU and CPU as the mainstream AV chip, especially for perception.

TI TDA3x is a DSP-based solution for AV applications. It has two floating-point DSP cores with vision AccleerationPac to accelerate the image processing performance. Each TDA3x also has a dual Arm, Cortex-M4 image processor. The TDA3x SoC processor enables ADAS algorithms such as autonomous emergency braking (AEB), lane keeps assist, advanced cruise control (ACC), traffic sign recognition, pedestrian and object detection, forward collision warning, and back over prevention. It is for entry-to-mid-segment automobiles with L2 and L3 levels [9]. DSP can process a large amount of data with digital signals. It uses a Harvard architecture, that is, the processor is connected to two independent memory banks via two independent sets of buses, allowing the fetching and executing instructions in parallel. One memory bank holds program instructions and the other holds data. The next instruction can be fetched and decoded while the previous instruction is executed. This architecture greatly increases the speed of the microprocessor. In addition, it also allows transmission between processing space and data storage space, thus increasing the flexibility of the device. It not only has programmability, but its real-time running speed can reach tens of millions of complex instruction programs per second, far exceeding that of general-purpose microprocessors. Powerful data processing capabilities and a high operating speed are the two most commendable features of DSP. Because of its strong computing power, fast speed, small size, and high flexibility in software programming, it provides an effective way to engage in various complex applications.

Mobileye EyeQ5 is an ASIC-based SoC solution for AV applications. Its basic architecture is a combination of MIPS CPU core and vector acceleration unit. The overall computing performance is 24 TOPS with only a 10 W power budget. The power consumption is the brightest spot and obvious advantage of using EyeQ5. EyeQ5 is designed based on a start-of-art 7 nm FinFET IC technology, and this chip

is aimed at L4 and L5 autonomous driving [10]. EyeQ5 is equipped with four heterogeneous fully programmed accelerators, which are optimized for proprietary algorithms, including computer vision, signal processing, and machine learning. EyeQ5 implements two PCIE ports at the same time to support multi-processor communication. This Compute architecture attempts to adapt the most suitable computing unit for each computing task. The diversity of hardware resources enables the fast operation of various applications and improves overall computing performance. However, the overall computing performance of EyeQ5 seems to be still far inferior to NVIDIA's solution.

2. Adopting self-designed, customized solutions

One customized solution is to take the x86 platform as the prototype, and directly integrate the Intel Xeon CPU and Nvidia's latest GPU architecture to achieve the highest computing performance. Another customized solution is to integrate GPU and FPGA to form a hybrid system by utilizing the benefits of short-latency, low power consumption, and high reliability of FPGA [11]. However, the disadvantage of those two solutions is that in addition to the need to customize the interfaces between CPU to GPU, and GPU to FPGA, heat dissipation, power distribution supply, sensor integration, functional safety, and connector interfaces are all required to be customized. Several companies like Tesla and Google/Waymo seek an edge in AV by making their own AI training silicon chips. Such a vertically integrated strategy for an AV company could be the ultimate solution for autonomous driving, which relies on deep neural networks (DNN) with huge computational demand. Google's Tensor Processing Unit (TPU) v3 is the latest ASIC-based AI accelerator mainly for DNN and machine learning [12]. It provides 420 TOPS computation performance for a single board. TPU is specially built for machine learning applications such as Google TensorFlow, which is designed to process more complex and powerful machine learning models in parallel at the price of reducing the accuracy of calculations. Compared to GPUs which are more suitable for machine learning and AI training, TPU is more suitable for analysis and decision-making after training. Tesla's D1 Dojo customized ASIC-based supercomputer chip can deliver 362 TOPS processing power [13]. Tesla places 25 of these chips on a single "training tile," and 120 tiles come together across several server cabinets (a total of more than an exaflop). The chip is built by a 7 nm semiconductor technology process and leaves the processor with an immense die size of 645mm^2, packing over 50 billion transistors.

How to choose the right Compute platform solution could depend on how to balance several metrics to achieve the best performance vs total life cost. According to Liangkai Liu et al. [14], 7 metrics shall be used to evaluate the computing system's effectiveness. They are accuracy, timeliness, power, cost, reliability, privacy, and security. The AV Compute platform integrates a variety of computing tasks with different attributes, such as precise geographic positioning and path planning, object recognition and detection based on deep learning, image preprocessing and feature extraction, sensor fusion and target tracking, etc. The performance and energy consumption ratios of these different computing tasks running on different hardware platforms are different. Generally speaking, for the convolution operation of

object recognition and tracking, GPU has better performance and lower energy consumption than DSP and CPU. For feature extraction algorithms that generate positioning information, DSP is a better choice. Therefore, to improve the performance and energy consumption ratio of the AV Compute platform and reduce the calculation latency, it is very valuable to adopt heterogeneous computing architecture. Heterogeneous Compute selects appropriate hardware implementations for different computing tasks, makes full use of the advantages of different hardware platforms, and shields hardware diversity through a unified upper-layer software interface. Table 2 shows the comparison of GPU, DSP, FPGA, and ASIC-based Computes for architecture, performance, power consumption, and cost. Adopting a self-designed SoC with customized AI training silicon chips could be the game-changer for the future AV industry. Figure 12 shows the comparison of different SoC platforms suitable for different load tasks.

Table 2 Comparison of GPU, DSP, FPGA, and ASIC-based computes

Boards	Architecture	Performance	Power Consumption	Cost
Nvidia Drive AGX	GPU	320 TOPS	300 W	$30,000
Xilinx Zynq UltraScale + MPSoC	FPGA	14 images/s/W	–	$1295
Tl TDA3x	DSP	–	30mW in 30fps	$549
Mobileye EyeQ5	ASIC	24 TOPS	10 W	$750
Google TPU v3	ASIC	420 TOPS	40 W	–
Tesla D1 Dojo	ASIC	362 TOPS	400 W	–
Qualcomm Snapdragon Ride L4/L5	ASIC	700 TOPS	130 W	–

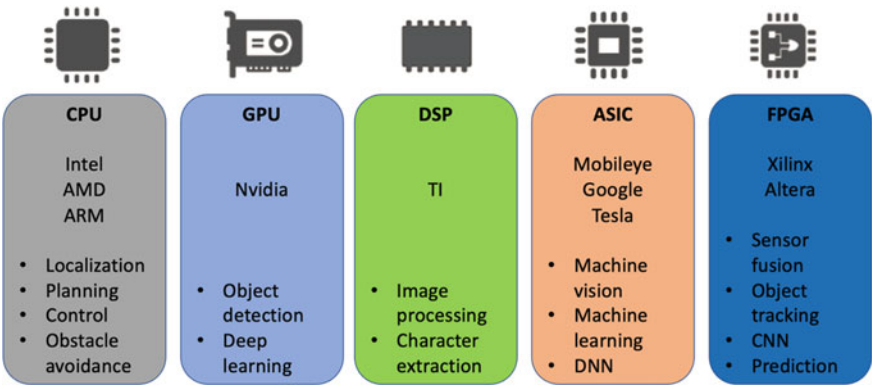

Fig. 12 Comparison of different SoC platforms suitable for different load tasks

3.2 *Environment Perception Sensors*

Sensors are to perceive the environment around the autonomous vehicle so AV can understand the environment correctly and make safe driving. To achieve the environmental perception, AV needs to obtain a large amount of surrounding information, specifically including the position and speed of surrounding vehicles, pedestrians, cyclists, and other moving agents, and possible behaviors of them at the next moment. AVs usually are equipped with cameras including IR camera, millimeter-wave radar, LiDAR, sonar, IMU, and GPS/GNSS to safely, accurately, and robustly collect such information as illustrated in Fig. 13. Moreover, typically there is more than one sensor of the same type. For example, to solve the blind spot and long-distance detection of LiDAR, both high-line-count radar, and low-line-count radar is generally used. As the horizontal viewing angle of a single camera is limited, multiple ($\geq$6) cameras are used to construct a 360° surround view. For millimeter-wave radar, due to the limitations of its horizontal viewing angle and distance factors, multiple radars are also used ($\geq$4). Furthermore, from a functional point of view, the redundancy between different types of sensors and from multiple same type sensors can improve the safety factor of the entire environment perception system. Perception generally is implemented by a chain of modules, comprising a sensor module, a microcontroller module, communication and networking infrastructure, and a Compute system. Perception needs to be robust and consistent across all use conditions. The requirements of perception are increasing with the increase in vehicle automation levels.

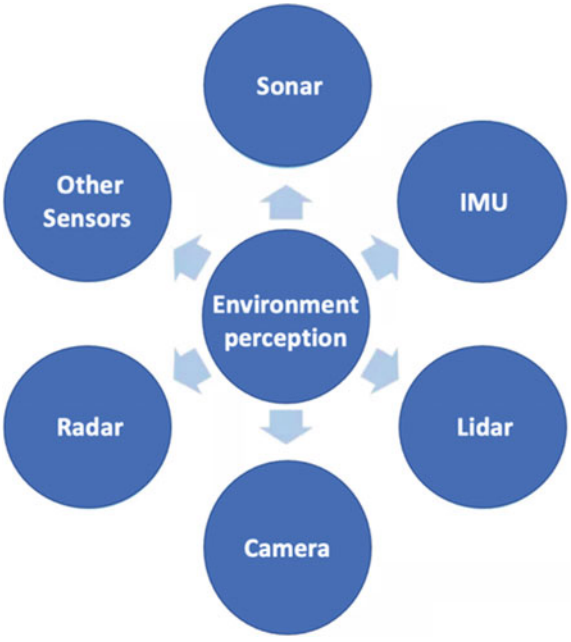

Fig. 13 AV environment perception hardware configuration

Technically speaking, first of all, the right sensing hardware needs to be chosen according to the needs of autonomous driving. This requires us to understand the advantages and disadvantages of different sensors. When sensors are available, we need to optimize the installation of these sensors to meet the needs of autonomous driving tasks. The environmental perception module is the most upstream of the automatic driving system. Through the analysis of data from different sensors, the analysis results are passed to the Compute module to realize the automatic driving of the vehicle. The sensing results of the environmental perception module include road dynamic and static target trajectories (such as vehicles, pedestrians, guardrails, etc.), traffic signal status (red, yellow, and green signal lights), traffic sign recognition related to traffic regulations, road lane line marking detection, and road surface detection.

According to the analysis of the output results of the environment perception module, we can get the key information related to the perception module task: 2D/3D target detection, scene semantic segmentation, instance segmentation, multi-sensor fusion, multi-target tracking, and trajectory prediction, as shown in Fig. 14. Although each technology can be designed independently, for the entire environment perception module, all different sensing technologies need to be fused to reduce the delay and memory storage consumption to achieve high efficiency, high precision, and low-cost objectives.

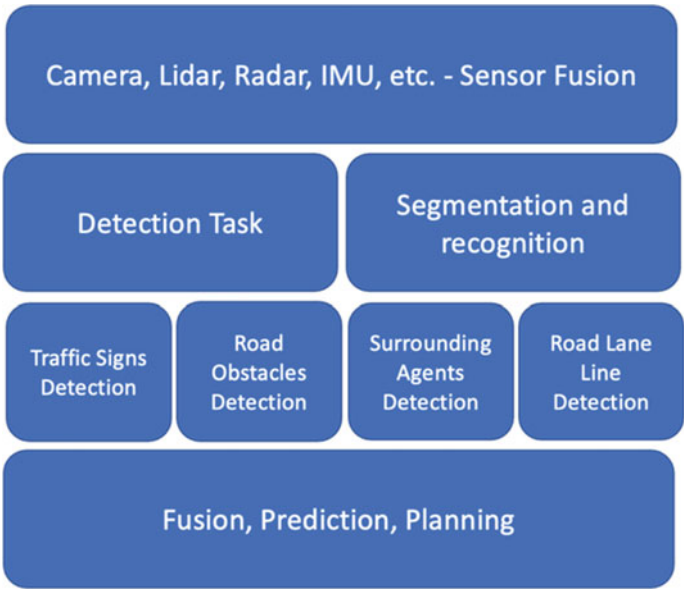

Fig. 14 Environment perception module task flow chart

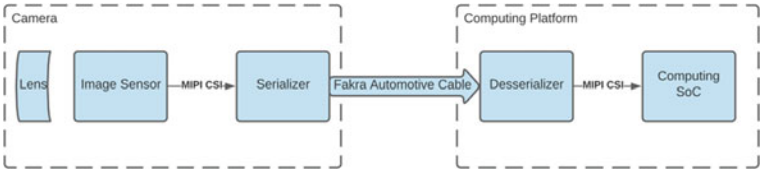

Fig. 15 The general camera block diagram and the data flow from the camera to the computing platform

3.2.1 Camera

Cameras are the most commonly used sensors to perceive the environment around the autonomous vehicle considering their relatively low cost and powerful usability. It is almost undisputedly adopted by all AV developers. The camera is the closest sensor type to the human eyes. The viewing range of the camera can vary from several centimeters to about 100 m. Also, they are often small, lightweight, and have low power consumption. The camera image provides a large amount of information at high frame rates, making it useful in tasks such as traffic light and pedestrian detection, lane tracking, object classification, traffic sign understanding, etc. Existing designs usually mount eight or more cameras around the vehicle to 360° to detect, recognize, and track objects. These cameras usually run at 60 Hz, making the total generated data a big challenge for Compute to real-time process such big data to get usable information about the environment. In addition, the quality of the camera's image is strongly affected by low lighting or bad weather conditions. The usability of the camera decreases significantly under heavy fog, rain, and snow. It is not good at long-distance vision as well.

In an AV, a camera in general consists of a Lens, image sensor, serializer, and power regulators. Different cameras may have different Lenses with different fields of view (FOV) and ranges. For example, 120-degree Len has a wide FOV but a short range. 30-degree Len has a narrow FOV but a long range. The image sensor is to detect and conveys information used to make an image by converting the variable attenuation of light waves into electronic signals. It has an active-pixel array-like 2MP or 8MP, and the Multiple Color filter array (CFA) like RGB Bayer, RCCB, etc. The Serializer is to convert the Mobile Industry Processor Interface (MIPI) Camera Serial Interface (CSI) to a single link. It can send the video frame data as well as receive the control data over the Fakra automotive cable as shown in Fig. 15.

3.2.2 Millimeter-Wave Radar

The radar is standard for Radio Detection and Ranging. It is a detection system to determine and calculate the distance and velocity using radio waves.

The Radar is mostly used to detect the distance to the objects around the vehicle. Once an object is detected too close to the vehicle, there may be a danger of collision

so the autonomous vehicle should take action as soon as possible. Examples of actions are braking or turning to avoid a potential collision. The data generated by the Radar is not needed to process too much. It will feed into the Compute directly. The Compute could implement the emergency action, such as an autonomous emergency brake. In the autonomous vehicle, the Radars are deployed in different areas such as the front, rear, front-right, and front-left of the vehicle. The front radars are typically mid and long-range radars responsible for autonomous emergency braking (AEB) and adaptive cruise control (ACC). The side radars are typically short-range radars to handle the requirements of blind-spot detection (BSD), front/rear cross-traffic alert (F/RCTA), and lane-change assist (LCA) [15].

Generally, 24 and 77 GHz frequencies are used in the radar system. 24 GHz includes industrial, scientific, and medical banks from 24 to 24.25 GHz. For 77 GHz, it has a 76–77 GHz band available for radar application. Compared to the 24 GHz frequency, the 77 GHz frequency has a wider bandwidth available, which improves the range resolution and accuracy significantly [15]. High range resolution results in better separations of objects. It also results in a better minimum distance detection. A shorter minimum distance is very important for some AV functions such as AEB. The higher frequency also can provide a better velocity resolution and accuracy. Another benefit of higher frequency is that the radar size can be made smaller. Radar can work under any weather conditions, which makes it indispensable. It has its unique capability to penetrate dust, fog, rain, and snow, therefore has a firm foothold on the AV sensor module.

3.2.3 LiDAR

LiDAR is the heart of object detection for most of the existing AVs. The full name of LiDAR is light detection and ranging or laser imaging, detection, and ranging. It can be used to calculate the distance. The difference between radar and LiDAR is that LiDAR has the laser generator and receiver inside. It sends millions of light pulses per second in a well-designed pattern to the surface of an object and measures the reflection time return to the receiver. With its rotating axis, it can create a dynamic, three-dimensional map of the environment. In an AV, the LiDAR is commonly used to detect objects and pedestrians, determine the distance, make high-definition maps, and localize a vehicle aligned with the high-definition map [16].

Compared to the Camera, LiDAR generates a 3-dimensional cloud image of objects instead of a 2-dimensional image. It has a larger sensing range, and the performance is less impacted by bad weather and a low lighting environment. Point cloud output from the LiDAR provides the data for autonomous computing to determine where objects exist in the environment and where the vehicle is in relation to those objects. It can generate a lot of data for vehicle Compute to process in real-time.

3.2.4 Ultrasonic Sensor

The ultrasonic sensor is a kind of radar that is widely used in vehicles already. It is often installed on the bumper at the rear, front, and sides of the car for the reversing assist and parking assist functions as shown in Fig. 16. Its working principle is to transmit high-frequency sound waves to gauge the distance between objects within close range. The ultrasonic sensor shows good performance in bad weather and a low lighting environment. But ultrasonic radar's maximum range is only about 20 m so it is not suitable for long-distance ranging. Ultrasonic radars can be used to complement other vehicle sensors, including radars, cameras, and LiDARs, to get a full picture of the immediate surroundings of a vehicle.

Ultrasonic sensors are generally composed of an ultrasonic transmitter, an ultrasonic receiver, a timer, a temperature sensor, etc. The distance measurement principle is to use the propagation speed of ultrasonic waves in the air to be known (344 m/s at 20 °C) and measure the sound waves in the air. After the launch, the time when the obstacle is reflected is calculated, and the actual distance from the launch point to the obstacle is calculated according to the time difference between the launch and the reception. It can be seen that the principle of ultrasonic ranging is the same as that of radar.

The formula of ranging is expressed as:

$$L = C \times T \tag{1}$$

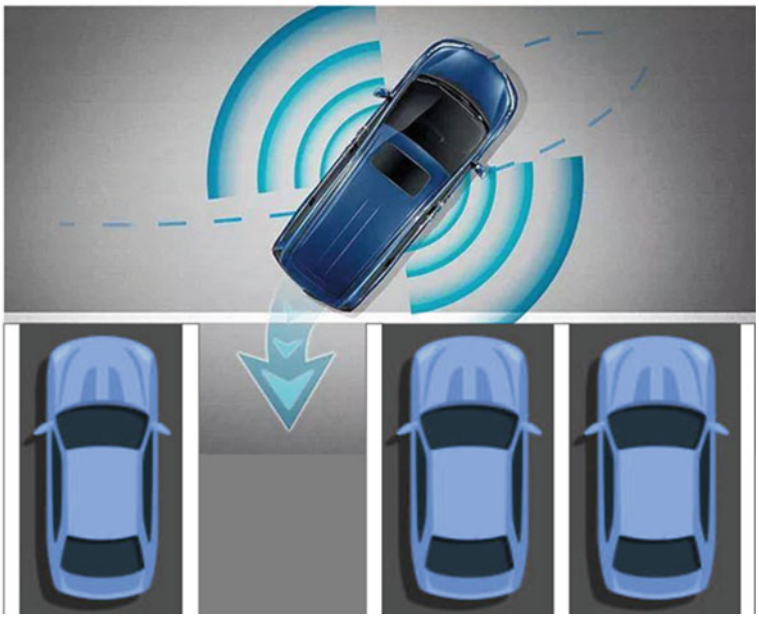

Fig. 16 Sonar-assisted parking illustration

where L is the measured distance length, C is the propagation speed of ultrasonic waves in the air, and T is the time difference of the measured distance propagation (T is half of the value of the time from emission to reception).

Table 3 illustrates a comparison of sensors, including camera, IR camera, radar, LiDAR, IMU, and ultrasonic sensors. The range of sensing distance for human eyes is 0–200 m. Human vision is poor during bad weather and low lighting condition. From the comparison, it shall be concluded that although humans have strength in the sensing range and show more advantaged functionality scenarios than any sensor, the combination of all the sensors can do a better job than human beings, especially in bad weather and low lighting conditions.

3.3 System on Chip (SoC)

A system on chip is a chip that integrates most components of a computer. The components consist of multiple cores of a central processing unit (CPU), graphics processing unit (GPU), artificial intelligence (AI) unit, multiple levels of cache, input/output ports of memory, high-speed I/O, internal connection between CPU, GPU, AI unit, memory, high-speed I/O, and the power management unit [17]. To support real-time data processing from various sensors, a powerful Compute is essential to AVs' success.

3.3.1 ASICS

In autonomous driving, Application-Specific Integrated Circuit (ASIC) consists of multiple units like the common CPU, the GPU, the unit for the deep learning, and the memory controller that connects the external memory through Low Power Double Data Rate 4 (LPDDR4). In general, different storages connect to the CPU. Flash is used to store the firmware. eMMC is for an application that needs more space. The UFS has a large size capability to store big data like a high-definition map. The camera data is transferred to the CPU through the Deserializer. The Deserializer converts the interface from Gigabit Multimedia Serial Link (GMSL) or Flat Panel Display Link (FPDlink) to the CSI. The LiDAR connects to the CPU through Ethernet Switch. The automotive 1GBase-T1 interface is from the Ethernet Switch. Sometimes, an ethernet physical layer (PHY) is needed to convert the 1GBase-T1 to other buses like Reduced Gigabit Media-independent Interface (RGMII) or Serial Gigabit Media-independent Interface (SGMII) if the Ethernet Switch can't support the 1GBase-T1.

The Micro Controller Unit (MCU) is used to manage the board. For example, monitor the health of the board like the voltage, current, and temperature, control the power on/off and reset. The radar data is going to the MCU through the Controller Area Network (CAN) bus. Ethernet Switch is used to communicate between CPU and MCU. The radar data is transferred to the CPU from MCU through the Ethernet Switch.

Table 3 A comparison of sensors, including camera, IR camera, radar, LiDAR, IMU, and ultrasonic sensors

	Advantages	Disadvantages	Detection distance (m)	Functionality
Lidar	High accuracy, wide detection range, 3D model of the surrounding environment, speed and distance estimates	Can be affected by bad weather such as rain, snow and fog. Less matured technology with high cost	200	Obstacle detection and recognition, road agents' speed and distance measurements, 3D model of surrounding environment
Camera	Identify the geometry and color of the objects. Recognize texts and symbols. Mature technology with low cost	Affected by changes in light, vulnerable to bad weather such as rain, snow and fog. Can't measure distance accurately	100	Obstacle detection and recognition, lane tracking, auxiliary positioning, road information understanding, map construction
Radar	Strong penetrating ability to smoke and dust, strong anti-interference, high accuracy of speed and distance estimates	Unable to apply visual recognition, such as size and shape of objects, Detection range is narrower than lidar	200	Obstacle detection—Medium and long distance
Ultrasonic sensor	Matured technology, low cost, strong antiinterference, less affected by weather	Poor measurement accuracy, small measurement range, short distance	3	Obstacle detection—Short distance, useful for BSM, parking assistance, and reversing assistance
IR/Thermal camera	Good vision at night or blind sun glare, reliable detect persons/animals	High cost	200	IR camera sees heat, reducing the impact of occlusion on classification of pedestrians
GPS/IMU	Localize vehicle position by combining satellite triangulation and inertial navigation	vulnerable to building and tunnel interferences, high cost	10	Localization

Figure 17 illustrates an ASIC-based Compute block diagram.

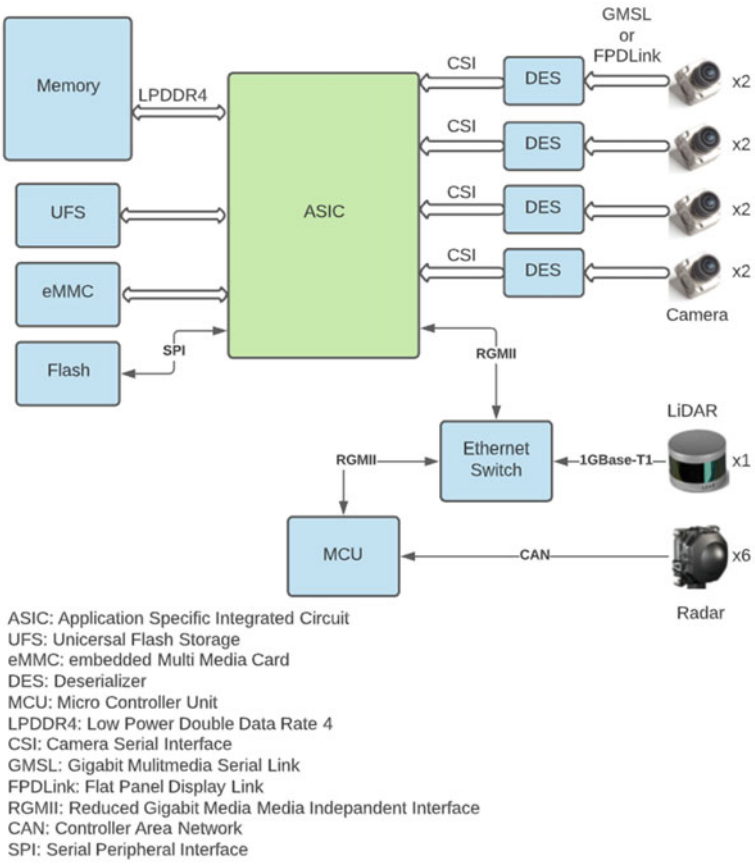

Fig. 17 ASIC-based block diagram

3.3.2 x86

x86 is a family of computer processing instruction set architectures (ISA) developed by Intel. ISA is computer architecture. It is an abstract model of a computer that defines the data types, registers supported, how to manage the memory and memory consistency, and how to access the input/output model. It also specifies the behavior of the machine code consisting of instructions. It is a low-level programming language used to control a computer processor [18].

The 8086 was developed in 1978 for 16-bit processors. Many additions and extensions have been added to the x86 instruction set over the years. In 1985, it grew to a 32-bit instruction set of the 80,386. The bit in both 32-bit and 16-bit is 32 or 16 binary digits. Today, an x86 microprocessor is used in almost any type of computer. It is also used as the computing platform in AV. In the computing platform used for AV, there are CPU and GPU. CPU performs basic arithmetic, logic, and control. The GPU is

more focusing on the artificial intelligence algorithm. Depending on the performance requirement, the computing platform can have a single CPU or dual-CPU solution.

In general, a single CPU solution has 1 CPU and GPU. CPU consists of the multiple cores, cache, memory controller which connects the memory devices, input/output controllers like the Peripheral Component Interconnect Express (PCIe) root complex which connects to the GPU and ethernet controller, and 10Gbps or 1Gbps ethernet outputs from ethernet controller. The camera, LiDAR, or radar data can be transferred to the CPU and GPU through the ethernet interface.

Platform Controller Hub (PCH) is Intel's signal chipset. It is the successor to the Intel Hub architecture that used two chips—northbridge and southbridge instead. It includes a clocking generator, PCIe interface, and storage interfaces like SATA and USB hub. The different storage devices can be connected by different interfaces such as the PCIe based hard disk or M.2 to PCH through PCIe and the SATA based hard disk or M.2 to PCH through SATA. The Direct Media Interface (DMI) is an interface that connects the CPU and PCH.

The firmware to perform the hardware initialization during the booting process is called the Basic Input/Output System (BIOS). It stores in the flash connected to PCH. It provides the runtime service for operating systems and programs.

The platform needs to be managed by monitoring the health of the system, controlling the power on/off, and resetting. It is done by the Baseboard Management Controller (BMC). The BMC has its memory, and flash with firmware. The 1Gbps ethernet to BMC can be used for remote access.

Figure 18 illustrates an x86-based Compute with a single CPU block diagram.

Besides the single CPU solution, to have more performance for the AI algorithm, there is a dual CPUs solution with dual GPUs. It can provide more CPU and GPU cores to increase workloads and performances. The communication between CPUs uses the high-speed interface Ultra Path Interconnect (UPI) to provide the high bandwidth between CPUs.

Figure 19 illustrates an × 86-based Compute with a dual-CPU block diagram.

3.4 Memory

The Synchronous Dynamic Random-Access Memory (SDRAM) is mostly used in the autonomous computing platform. The read and write operation is through an interface synchronous with the system bus. The data and control signals are aligned with the clock signal.

There are different standards of the SDRAM such as Single Data Rate (SDR) SDRAM and Double Data Rate (DDR) SDRAM. SDR reads/writes one time in one clock cycle. DDR SDRAM is the next-generation of SDR SDRAM. The data can be transferred two times in one clock cycle, at the rising and falling edges of the clock signal. Thus, it achieves higher bandwidth as compared with the SDR. It doubles the data rate without increasing the frequency of the clock.

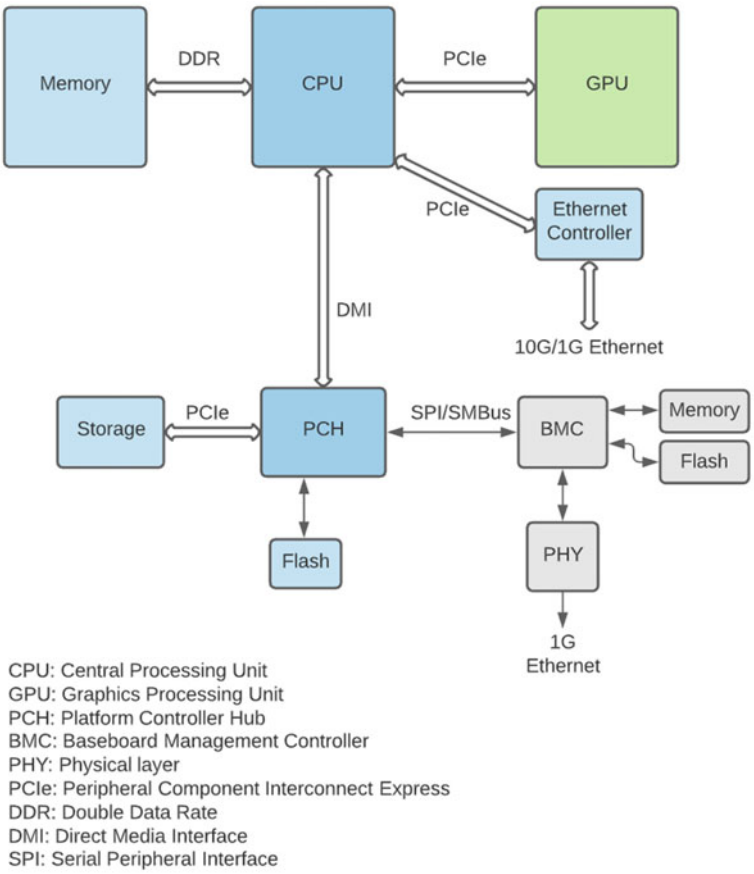

Fig. 18 x86 block diagram with single CPU

DDR2 SDRAM has a data rate twice as fast as DDR SDRAM. It is achieved by doubling the prefetch buffer to 4 bits. The DDR3 SDRAM has an 8-bit prefetch buffer. As a result, the data rate doubles based on the DDR2. The DDR3 SDRAM reduces power consumption by lowering operation voltage. DDR4 SDRAM lowers its operating voltage. It adds four new back groups to achieve a higher data rate. Table 4 compares different SDRAM.

DDR SDRAM is mostly used in the x86 platform. In the ASIC-based platform, the Low Power Double Data Rate (LPDDR) like LPDDR4 is used to save power.

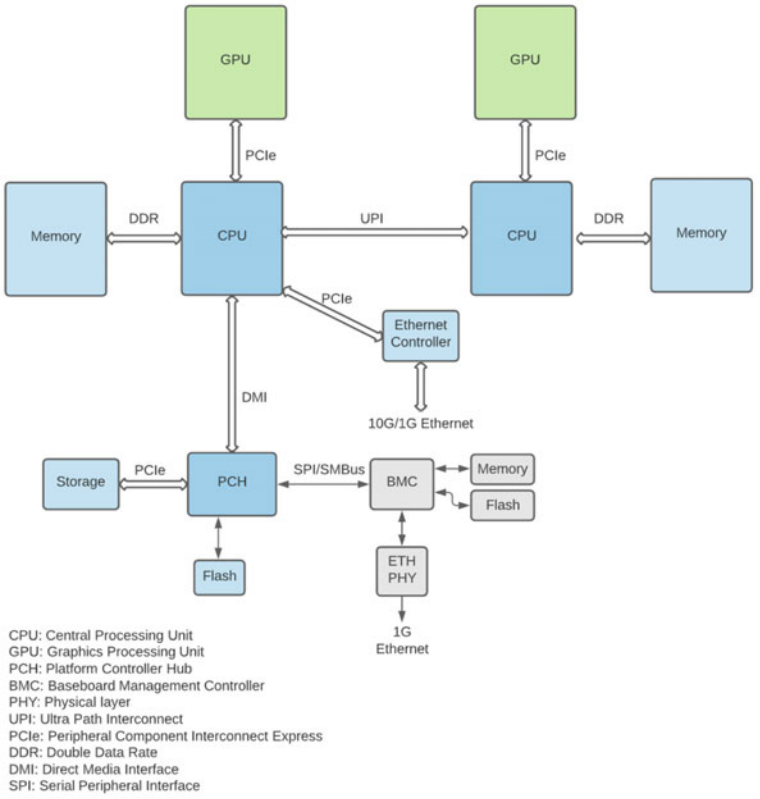

Fig. 19 x86 block diagram with dual CPUs

Table 4 Comparison of different SDRAM

SDRAM standard	Internal data rate (MHz)	Interface clock data rate (MHz)	Prefetch	Interface data rate (MT/s)	Operation voltage (V)
SDR	100–166	100–166	1*n*	100–166	3.3
DDR	133–200	133–200	2*n*	266–400	2.5
DDR2	133–200	266–400	4*n*	533–800	1.8
DDR3	133–200	533–800	8*n*	1066–1600	1.5
DDR4	133–200	1066–1600	8*n*	2133–3200	1.2

3.5 Storage

In autonomous driving Compute, there are different kinds of storage devices for different purposes. For example, Serial Peripheral Interface (SPI)/Quad Serial

Peripheral Interface (QSPI) Flash, Embedded Multimedia Card (eMMC), Universal Flash Storage (UFS), and Solid State Drive (SSD) are commonly used.

SPI/QSPI Flash is to store the SoC firmware. During the system boot-up, to initialize components inside the SoC as well as provide the runtime service for the operating system and programs, the SoC loads and executes the firmware from this Flash through SPI/QSPI bus.

eMMC is similar to the SPI/QSPI Flash. The SoC firmware can be stored in it. Generally, eMMC has a larger capability. Except for the firmware, it can store the operating system and applications running in the operating system.

UFS has a larger capacity and higher bandwidth with a differential signal interface. Except for storing the SoC firmware, it can store the high-definition map as well as the training data for autonomous driving Artificial Intelligence Algorithm.

In autonomous driving, the data captured from the cameras, radar, and LiDAR is huge. Collecting all the sensors data as much data as possible will help to train the AI model. The SSD has a big capacity. It normally is used to collect the data captured from camera, radar, and LiDAR.

For AV applications, memory must have enough read and write endurance to match the excessive data logging requirements of a vehicle over its lifetime. Consider an SSD with an endurance of 10^6 accesses before a cell typically degrades. At a record rate of 0.2 s, an SSD block would wear out in less than three days. To extend the effective endurance of SSD, wear-leveling has been used. Wear-leveling involves tracking the reliability of each memory block and moving data to a new block when the current block begins to experience errors beyond a certain threshold.

3.6 Network

In the autonomous vehicle, mainly two networks are used. One is the ethernet network. The communication between the controlling process and the computing process is through this network. LiDAR data is captured and transferred through an ethernet network. In the x86 platform, the Camera data also uses the ethernet to transfer to CPUs. It offers high bandwidth interfaces like 1 or 10 Gbps. Normally, the automotive-grade Ethernet Switch chip and the media convertor physical layer (PHY) are used in the Ethernet network.

The other network is the controller area network (CAN). It is mainly used in the communication between electronic control units (ECU). Radar data is captured and transferred to the Compute through the CAN network.

With the development of AD technology, more and more data need to be transferred between computing processer, control processor, and different ECUs. More Camera, LiDAR, and Radar data are captured and transferred. The bandwidth of the network inside the vehicle becomes more and more important. The ethernet network will be more popular in the AD platform.

3.7 Real-Time Operating System

To make AD safe, perception, prediction, and deciding in real-time are important. So, a real-time operating system (RTOS) is used in an AV. RTOS is fast and responsive. It is intended to run a real-time application. The RTOS is mainly used in many embedded systems. It requires real-time processing. Due to the hardware resource, performance and efficiency are high priorities. The scheduler in an RTOS is designed to provide a predictable (normally described as deterministic) execution pattern. This is useful for embedded systems.

RTLinux and QNX are two popular RTOS systems. RTLinux runs all the operation system (OS) components in the kernel space, including memory management, file management, networking, and drivers. It can improve performance. It also can respond faster and more reliably. The downside is that since all the components are in the kernel space, a single failure can cause the OS to crash [19].

Compared with RTLinux, QNX has a core RTOS kernel to access the whole system. It allocates the memory for other processes. All the other components run in their own isolated space. It improves reliability and security. Also, it isolates the error in one component from other components.

3.8 Management, Failure Detection, and Diagnostics

Safety is very critical in an autonomous vehicle. To make the vehicle safe, failure detection, diagnostics, and platform management become impotent. There are different kinds of failure detection to cover the autonomous computing platform. The voltage, current, and temperature monitoring is the hardware-level fault detection mechanism. Run time diagnostics like CPU internal self-test, memory bit error detection, and storage bit error detection is important to detect and report any failure that happened.

The MCU in the ASIC-based platform, as well as the BMC in the x86-based platform, are mainly used to detect the failure, manage the power on/off, and reset other domains like the computing domain, network domain, camera domain, etc. They run diagnostics applications to monitor critical functions.

3.9 Security and Middleware

For AV, security is very important. It is extremely dangerous for any AVs to get on the road without meeting the rigid security requirements. At present, there are a variety of methods for AV to be attacked and the attacks can happen at every level of the AD system, including sensors, Compute system, control system, and vehicle networking communication system. First of all, the attack on the sensors does not need to enter

the AD system. Therefore, the technical threshold of this external attack method is quite low, that is, it is simple and direct. Second, if hackers enter the AD system remotely, they can crash the system to cease the vehicle operation. They also can directly steal sensitive vehicle information. Third, if hackers enter the AV control system, they can directly manipulate and control the mechanical components so that they can hijack the vehicle to make terroristic attacks, which is extremely dangerous. Fourth, the Internet of Vehicles links different AVs and the central cloud platform system. Hijacking the Internet of Vehicles communication system can also cause communication chaos in the AVs. Therefore, car interconnection through V2V and V2X can bring great convenience to users, but it also exposes vehicle systems to the risks brought by the internet. The security requirements of AV become more and more challenging due to:

- More and more networked and intelligent vehicle controllers used: BCM, IMMO, PKE/RKE, TBOX, IVI, ADAS, etc.
- More and more networked and intelligent vehicle sensors are used: TPMS, Camera, LIDAR, RADAR, etc.
- More and more input ports, interface layers, and codes used: OBD, CAN, wireless, mobile phones, cloud, etc.
- More and more cloud control, AV remote control used: remote management, frequent OTA, remote driving, remote mobile phone control, etc.
- More and more vehicle communication protocols are used: 4G/5G, Wi-Fi, Bluetooth, NFC, RFID, etc.

The automotive security categories can be classified as component/sensor security, network security, and control security as shown in Table 5. The sensor security includes jamming or spoofing the sensors like Cameras, Radars, LiDARs, and GPSs. Network security includes attacking the network and sending the wrong message to the network. Multiple services are running in the autonomous driving system. To facilitate the dependencies between the services. The middleware is impotent to simplify the communication between different autonomous driving services. It is on top of the RTOS.

SAE's J3061 procedure "Cyber-physical Convergence System Cyber Security Guidelines" released in January 2016 is the first guidance document formulated for automotive cyber security. The supporting document J3101 "Hardware Protection Safety Requirements for Road Vehicle Applications" allows designers to take some measures to provide multiple protections for vehicles, such as storing the verification key in the protected area of the microcontroller. For AV, safety, and security, in general, are considered the top items in the development of AD technology. To reduce and avoid the risks in actual road operation, adequate simulation, bench, and closed field testing and verification must be done before actual road deployment.

Table 5 Classification of component and system-level security

Security category		Security content
Component security		Authentication protocols such as verification key Secure start and communication Security certification and upgrade Security monitoring Embedded with TEE and HSM Intrusion detection system Hardware root of trust
Vehicle information system security	In-vehicle network security	Sub-networks Gateways Visit control Protocol encryption and authorization Abnormal vehicle control detection
	OS and software control security	Anti-flash FW Prevent the denial of service and attacks Anti-sniffing Protocol authorization and management Data encryption

4 Electrical Functional and Reliability Validation

The automotive industry is a highly regulated industry across the globe. To survive in the market for a long period, automotive OEMs and component manufacturers need to be constantly innovative in terms of quality, durability, reliability, and safety. They also need to ensure that the system and components of the automobiles must function properly throughout their working life. With the fast-growing innovations in the industry such as EV and AV, new testing solutions and methodologies are constantly needed accordingly. From a testing and validation perspective, AV Compute brings together two previously separate validation standards: the automotive industry standards such as GMW3172 and ISO16750 standards, and the electrical industry standards such as IEC and JEDEC standards. The benefits of conducting automotive level electrical functional and reliability validation are (1) ensure the electrical safety of users during the product operation, (2) verify that the products comply with the state-of-art industry standards, (3) evaluate the conformance, interoperability, and electromagnetic compatibility, (4) validate product durability and reliability along with the cost of the warranty. Functional and reliability tests are the ways to identify manufacturing faults and design weaknesses that could compromise the electrical safety and durability of a Compute out in the field. Thorough functional and reliability tests protect against the risk of safety and reliability issues so that Compute

can be used for its intended purpose with minimal chance of accidents and failures occurring.

In general, electrical AV electronics could present significant challenges for automotive testing. High currents and voltages are present in the form of complex signals both as stimuli and as measurements. Much of the circuitry involves asynchronous timing and events. In this section, we will introduce Compute electrical functional test and reliability validation based on GMW3172 and ISO16750. GMW3172 and ISO16750 are automotive industry well-established and accepted electronic components testing standards. Those standards have been used to systematically qualify electronic components for the life cycle of all GM and other vehicle OEM manufactured vehicles with a set of testing environmental conditions and pass/fail requirements. In the process of forming the standards, various environmental factors, world climate conditions, vehicle types, vehicle operating conditions and working modes, product life cycle, vehicle power supply voltage, and component installation locations in the vehicle were taken into consideration. We are going to describe the overall Compute validation testing in three categories: EE functional testing, reliability validation testing, and EMC/ESD compliance testing.

4.1 Automotive Level EE Functional Tests

4.1.1 Five-Point Functional/parameter Check

For fully functional/parameter testing, a 5-point check is required. This test is to let Compute be exposed to three temperatures and three voltages. The operating types are 2.1 defined by GMW3172, Compute functions are not activated to confirm functionality in sleep mode/off mode, and 3.2 defined by GMW3172, Compute with electric operation and control in typical operating mode. The five points are defined as:

1. $T_{\min}$, $V_{\min}$
2. $T_{\min}$, $V_{\max}$
3. T_{room}, V_{nom}, where U_{nom} is V_{B} for Operating Type 2.1, and U_{A} for Operating Type 3.2
4. $T_{\max}$, $V_{\min}$
5. $T_{\max}$, $V_{\max}$

The test condition for this test is:

(a) Step 1:

Test temperature for Chamber: $T_{\min}$
Testing time: 75 min
Operating type: 3.2
Test voltage: 9 VDC and 18 VDC

(b) Step 2:

Test temperature for Chamber: 23 °C (Room Temperature)
Testing time: 75 min
Operating type: 2.1 for 12 VDC test voltage -> 3.2 for 14 V test voltage
Test voltage: 14 VDC

(c) Step 3:

Test temperature for Chamber: $T_{\max}$
Testing time: 75 min
Operating type: 3.2
Test voltage: 9 VDC and 18VDC.

4.1.2 One-Point Functional/Parameter Check

One-point functional/parametric check is to verify Compute full functionality under one single temperature and one single voltage condition. It is a special case of the 5-point check. The 1-point check shall be performed at room temperature under a nominal voltage unless otherwise specified. The temperature shall be stabilized before the 1-point Functional/Parametric Check.

4.1.3 Continuous Monitoring

Continuous monitoring shall monitor the functional status of the Compute during the test environment continuously. Continuous monitoring shall record all input and output signals, serial data messages, all transmitted packets, voltages, frequencies, powers, and temperatures from all critical components, and erroneous Input/Output (I/O) commands or states.

4.1.4 Electrical Load Testing

In Table 6, selected electrical load testing items are listed specifically for Compute used for battery-powered AV.

4.2 Reliability Validation Tests Based on AV Mission Profiles

The reliability of AV hardware, especially the Compute, is one of the critical enablers for AV business. Reliability is defined as the probability that a product will perform its required function for a given time at the desired confidence level under the specified use conditions. The failure of a Compute is defined as the termination of the ability of the Compute to perform a required function. The function usually is specified

Table 6 List of electrical load tests for compute

Test item	Standards	Purpose	Requirement
Direct current supply voltage	ISO16750-2	Validate Compute functionality at minimum and maximum input voltages	All functions of the device/system perform as designed during and after the test
Overvoltage	GMW3172	Verify Compute immunity to overvoltage conditions	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test
State change waveform characterization	GMW3172	Verify that the Compute behaves adequately during state changes (e.g., Compute cold start, shutdown, etc.)	All functions of the device/system perform as designed during and after the test
Reverse polarity	GMW3172	Check the ability of the Compute to withstand against the connection of a reversed battery in case of using an auxiliary starting device	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test
Jump start	GMW3172	Verify the Compute's immunity to positive overvoltage. This condition can be caused by a double-battery start assist	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test
Slow decrease and increase of supply voltage	ISO16750-2	Simulate a gradual discharge and recharge of the battery	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test
Ground reference and supply offset	ISO 16750-2	Verify reliable operation of the Compute if two or more power supply paths exist. For instance, a component may have a power ground and a signal ground that are output on different circuits	All functions of the device/system perform as designed during and after the test

(continued)

Table 6 (continued)

Test item	Standards	Purpose	Requirement
Parasitic current	GMW3172	Verify that the compute's power consumption complies with the specification for Ignition OFF state. This is to support power management and engine start ability following long-term storage and parking conditions	The maximum allowable average parasitic current shall be 0.125 mA. Analyze the stored current waveforms for any random fluctuations. Unintentional wakeups are not allowed
Power supply interruptions	GMW3172	Verify the proper reset behavior of the compute. This test shall also be used for all microprocessor-based components to quantify the robustness of the design to sustain short-duration low voltage dwells	All functions of the device/system perform as designed during and after the test
Battery voltage dropout	GMW3172	Verify the compute's immunity to voltage decrease and increase that occur during discharge and charging of the vehicle battery	There shall be no inadvertent behavior during the transitions. Different functional statuses are required pending on the zone
Pulse superimposed voltage	GMW3172	Verify the compute's immunity to supply voltage pulses that occur on battery supply in the normal operating voltage range	All functions of the device/system perform as designed during and after the test
Intermittent short circuit to battery and to Ground for I/O	GMW3172	Verify the Compute's immunity to intermittent short circuit events on Input/Output (I/O) lines as well as the component's ability to recover automatically from these events	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test
Continuous Short circuit to battery and to ground for I/O	GMW3172	Verify the Compute's immunity to continuous short circuit events on Input/Output (I/O) lines	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test

(continued)

Table 6 (continued)

Test item	Standards	Purpose	Requirement
Open circuit—Single-line interruption	ISO16750-2	Verify that the Compute is immune to single-line open circuit conditions	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test
Open circuit—Multiple line interruption	ISO16750-2	Ensure functional status as defined in the specification of the Compute when the Compute is subjected to a rapid multiple line interruption	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test
Ground offset	GMW3172	Verify the Compute's ability to function properly when subjected to ground offsets	All functions of the device/system perform as designed during and after the test
Discrete digital input threshold voltage	GMW3172	Verify the capability of discrete digital input circuits (including switch interfaces) to withstand minor voltage fluctuations without causing a change of active/inactive state	All discrete digital input interfaces shall be able to correctly detect the logic levels
Over load—All circuits	GMW3172	Verify the component's ability to withstand overload situations or open circuits in a safe manner	If an output is over-current protected: one or more functions of the component do not perform as designed during the test and do not return to normal operation after the test until the component is reset by any "operator/use" action If an output is not over-current protected: one or more functions of the component do not perform as designed during and after the test and cannot be returned to proper operation without repairing or replacing the component

(continued)

Table 6 (continued)

Test item	Standards	Purpose	Requirement
Insulation resistance	GMW3172	Verify the component's immunity to loss of insulation	One or more functions of a device/system do not perform as designed during the test but return automatically to normal operation after the test The insulation resistance shall be greater than 10 MΩ
Power offset	GMW3172	Verify the component's ability to function properly when subjected to power offsets	All functions of the device/system perform as designed during and after the test

in Compute technical specification or operation manual. Failures could be the loss of whole or partial functions either permanently or intermittently, the deterioration of the whole or partial function over time, and a field surprise. The purposes of performing reliability validation tests on the Compute are to quantify the factors limiting the life of a Compute significantly less than the total expected life and to provide guidelines for design for reliability (DfR) and field replacement. There are two important concepts generally involved with reliability validation tests, the acceleration concept and the statistics concept. Usually, the products last so long that their lifetime can't be verified by direct measurement, therefore, accelerated tests, as well as the extrapolation procedures, are mandatory for reliability engineering. On the other hand, the reliability test is a sampling test, which is not testing the entire population. Thus, the true probability of failure can't be obtained. The probability of failure of a population can only be inferred. As a result, the concepts of uncertainty and confidence arise from the fact that it can test only a limited sample from a large population. The statistical theory for reliability such as the reliability function, the probability density function, the cumulative density function, the hazard rate, the conditional reliability function, and mean time to or between failure (MTTF or MTBF) are needed.

In general, two different reliability validation approaches are used in the automotive industry, the knowledge-based approach and the standard-based approach. For the knowledge-based approach, the failure rates are quantitatively determined under various use conditions based on DFMEA, physics of failure models, continuous probabilistic model, and prior knowledge with customized stress conditions. Usually, failures are needed so are good. In contrast, the standard-based approach proves that a defined failure rate is met based on specifications, experience, and shipped product field return knowledge. It is a zero-failure test or test of Bogey. N parts are tested to one life and no failures are allowed so failures are bad. The mathematical interpretation is that the product has unknown inherent reliability, R. The reliability test verifies that R exceeds a critical value with a specified probability or

confidence, C, as shown in Eq. 2.

$$\Pr(R > R_{\text{critical}}) = C \tag{2}$$

For the automotive industry, standard-based validation has been popularly used. Many standards such as AECQ, GM3172, ISO16750, JEDEC, IEC, etc. form a framework throughout the industry for easy implementation. It is a simple digital "pass or fail" method. It requires a sample of a predetermined size to be tested for a specific length of time under a specific test condition. The required reliability then is demonstrated if no failures occur at the end of the test. In this section, we will introduce a set of standard-based reliability validation tests based on GMW3172 and ISO16750 standards.

4.2.1 AV Mission Profiles

Automotive electronic component reliability validation tests start and end with the mission profile. When specifying a component, it is common for OEMs and their suppliers to develop a specific mission profile, which is essentially a summary of all the expected environmental and functional conditions that the component will face during its service life. As AV largely will be used for Robo-taxi rideshare service, the fleet can be deployed at a specific location following its unique operational design domain (ODD). Furthermore, it can be controlled 100% by service providers. Therefore, it will have its customized mission profiles to mimic a particular type of field stress, as well as its related severity. In addition to the operation life mission profile that was discussed in the introduction section, the most commonly referenced stresses are related to temperature, humidity, dynamic loads, thermomechanical stress, chemical load, UV radiation, dust ingression, water ingression, and EMC load.

The customized mechanical random vibration mission profile is usually obtained by installing a series of accelerometers at various vehicle locations to record the transfer function through the vehicle operation. As shown in Fig. 20, an example of dynamic responses from a vehicle driving at Gomentum Station proving ground located in Concord, California to mimic smooth suburb and city driving roads was recorded [20]. The solid colorful wavy lines are acceleration power spectral density (PSD) curves instrumented on the roof rail of a vehicle from different runs, the solid black line is the envelope profile, the dash black line is the margin profile, and the solid red line is the accelerated profile.

The customized temperature mission profile is usually obtained by installing a series of temperature loggers at various vehicle locations to record the temperatures through the vehicle operation. As an example of the temperature profiles of San Francisco shown in Fig. 21, roof temperature and trunk temperature are dependent on ambient air temperature, vehicle driving speed (airflow), solar loading, and heating sources from the surrounding components. The highest peaks are associated with vehicle idles. The trunk temperature has less temperature swing as compared to the

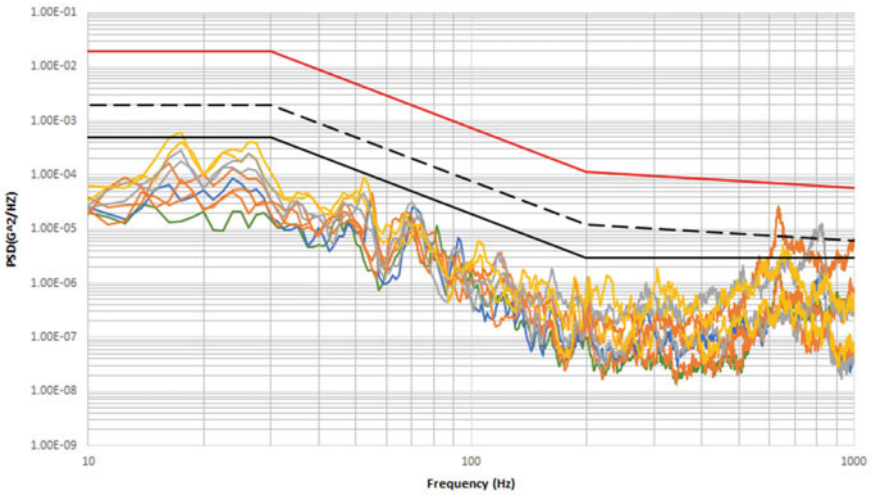

Fig. 20 Example of customized random vibration profile

roof temperature. By comparing the CDFs of roof and trunk temperatures as shown in Fig. 22, both of them exhibit a multimodal distribution.

Customized temperature cycling (TC) mission profiles can be inferred from the time-dependent temperature loggings. Endo and Matsuishi [21] developed the Rainflow Counting (RFC) method by relating stress reversal cycles to streams of rainwater flowing down a Pagoda. The rainflow counting algorithm is one of the popular counting methods used in fatigue and failure analysis from a time history for cycle counting and was adopted as a standard by ASTM E 1049-85. The rainflow counting method allows the application of Miner's rule to assess the fatigue life of a structure

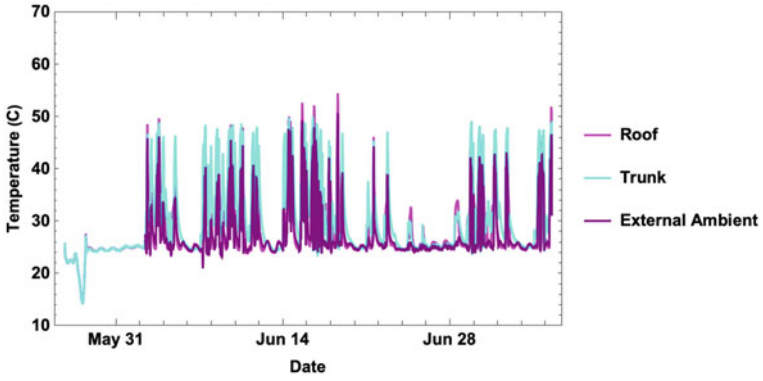

Fig. 21 Temperature time-dependent profile for roof and trunk locations together with the external air ambient temperature

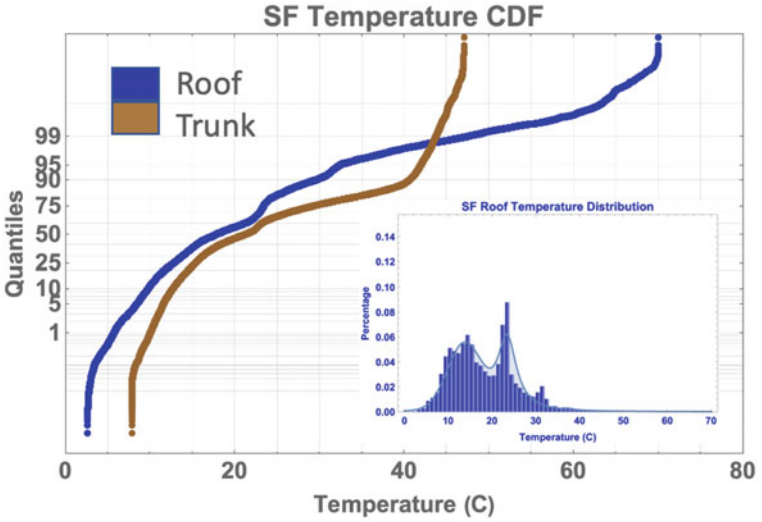

Fig. 22 CDF of roof and trunk temperatures. Inset: PDF of root temperature

subject to complex loading. For TC mission profile establishment, RFC is recommended to avoid potential over-stress and under-stress TC damages. Figure 23 shows an example of computing operating temperature in the trunk recorded up to 1600 h and calculated dT and average temperature distribution based on RFC. Accordingly, based on Miner's rule of linear accumulation of the damage, when the damage fraction (LC) reaches 1, failure occurs per Eq. 3, the effective dT, therefore, can be determined for stress to field condition transformation.

$$\text{Total Damage} = \sum_{i}^{m} \left(\frac{\Delta T_{\text{stress}}}{\Delta T_{\text{ref}-i}} \right)^{n} \times P_i = \left(\frac{\Delta T_{\text{stress}}}{\Delta T_{\text{ref}-\text{eff}}} \right)^{n} \tag{3}$$

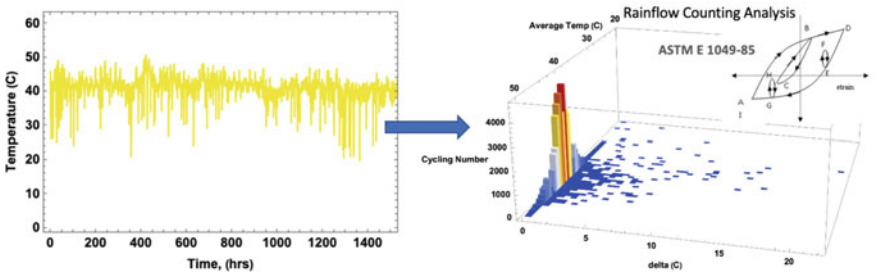

Fig. 23 An example of Compute operating temperature in the trunk, and calculated dT and average temperature distribution based on RFC

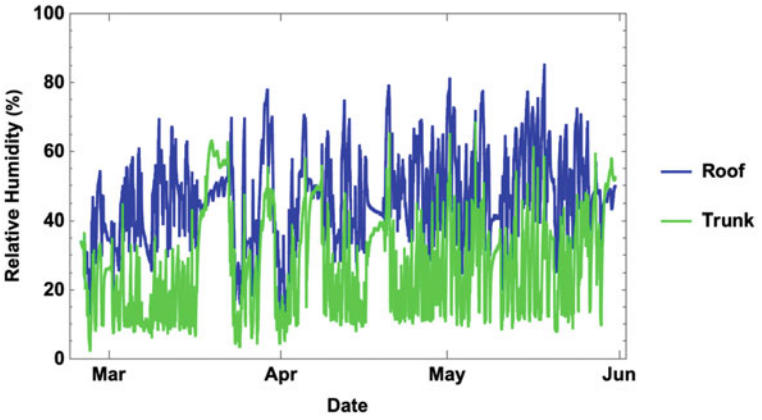

Fig. 24 Time-dependent RH levels from roof and trunk recorded during vehicle operation in San Francisco

where ΔT_{ref} is the reference temperature change from the stress condition, and ΔT_{eff} is the effective temperature change for field operation derived from TC mission profile data and Miner's rule.

The customized humidity mission profile is usually obtained by installing a series of humidity loggers at various vehicle locations to record the relative humidity (RH) through the vehicle operation. Figures 24 and 25 show the time-dependent relative humidity level from roof and trunk locations during vehicle operation in San Francisco, and the CDF of those RH values. Interestingly, by plotting RH vs temperature as shown in Fig. 26, it was found that high-temperature high humidity conditions likely can't co-exist even in a coastal city like San Francisco. Also, it was found that the location is important. Vehicle trunk location is much drier with less RH change than roof location.

Lastly, a customized solar loading mission profile is usually obtained by installing a set of pyranometers at various vehicle locations to record the solar intensity through the vehicle operation. Figure 27 shows an example of solar intensity in downtown San Francisco on a day in September 2018. It indicates that solar loading has a strong dependence on vehicle speed, and location (indoor vs. outdoor). Furthermore, it was found that component surface finish color has a profound impact on component temperature under solar loading as shown in Fig. 28. Overall, we found solar load effect has geometric location, seasonal, daytime, surface finish, indoor vs. outdoor, and the state of the atmosphere dependences.

4.2.2 Reliability Validation Tests

Reliability is a method to determine how long a product will last. Therefore, reliability engineering is the prediction of the life of the product. It is different from quality

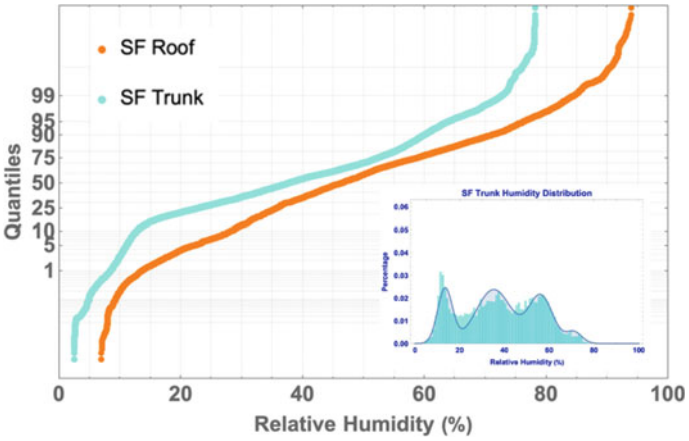

Fig. 25 CDF of RH values shown in Fig. 28

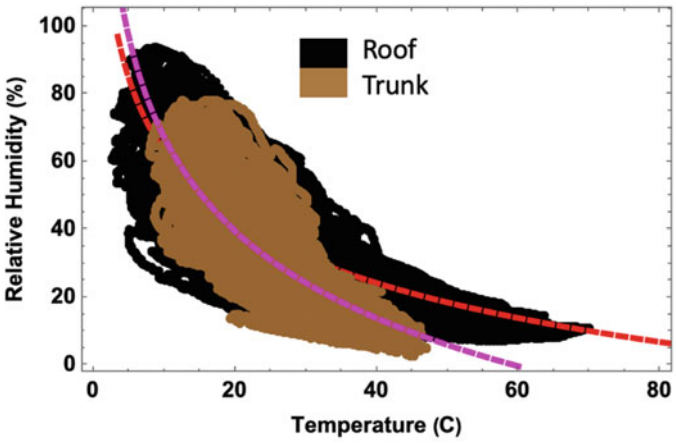

Fig. 26 An inverse relationship between RH and temperature for both roof and trunk locations

which means how closely the product meets user needs. Reliability testing is the testing of the life of the product by repeatedly making the product go through the stresses for an estimated duration or number of cycles and checking for failures.

As we mentioned earlier, reliability testing is sampling testing. The probability of failure of a population can only be inferred. Thus, the concepts of uncertainty and confidence arise. Practically, we are dealing with two kinds of confidence during our work. The first one is engineering confidence, which is mostly a matter of judgment and experience based on people. The second one is statistical confidence, which is used to make inferences about a population based on the sampling data. Statistical confidence is the one that directly impacts the reliability testing plan. A good reliability testing plan should be able to build a statistical sample size, meet a particular

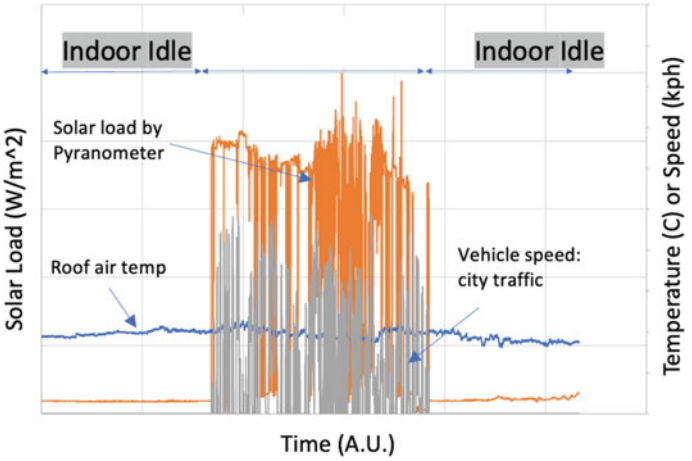

Fig. 27 Solar intensity recorded by a pyranometer in downtown San Francisco

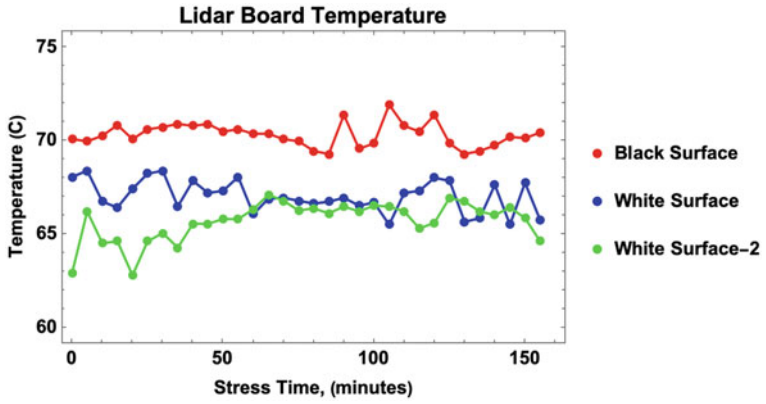

Fig. 28 LiDAR board temperatures with different surface finishes

reliability objective or goal, and achieve a specific confidence level. In general, two statistical approaches can be used for developing a reliability testing plan for sample size N, the chi-squared testing approach and the Weibull Bayesian estimate of zero-failure approach. The chi-squared testing approach is for the flat part of the failure rate bathtub curve or random failures with a constant failure rate as shown in Eq. 4:

$$N = \frac{\chi^2(\alpha, 2n+2)}{2\overline{\lambda} \times \mathrm{AF} \times t} \tag{4}$$

where $\overline{\lambda} = 1/\mathrm{MTBF}$ and is an upper bound failure rate objective, AF is acceleration factor, n is a number of failures, and α is confidence level.

Weibull Bayesian estimate can model the entire bathtub curve with different Weibull slope values as shown in Eq. 5:

$$N = \frac{Ln[1-\alpha]}{Ln[R] \times \left(\frac{t_{\text{test}} \times \text{AF}}{t_{\text{spec}}}\right)^{\beta}} \tag{5}$$

where R is lower bound reliability objective, β is the Weibull slope with $\beta < 1$ for early life failures, $\beta = 1$ for random failures, and $\beta > 1$ for wear-out failures, t_{test} is the total test time, and t_{spec} is the specification life. Weibull Bayesian estimate can't allow any failures.

Different acceleration factors are used for different failure mechanisms to determine the sample size or testing duration per different reliability tests. For thermal shock, temperature cycling, and power temperature cycling, a modified Norris-Landzberg model or simple Coffin-Manson model could be used. Modified Norris Landzberg consists of four parts as shown in Eq. 6.

$$\text{AF} = \left(\frac{\Delta T_{\text{test}}}{\Delta T_{\text{field}}}\right)^{a} \times \left(\frac{\text{Dwell}_{\text{test}}}{\text{Dwell}_{\text{field}}}\right)^{b} \times \left(c \times \text{RampRate}^{d}\right) \times \text{e}^{f \times \left(\frac{1}{T_{\text{fieldmax}+273}} - \frac{1}{T_{\text{testmax}+273}}\right)} \tag{6}$$

The parameters of Norris Landzberg for lead-free solder and lead solder are listed in Table 7 for reference. The first part in Eq. 6 is the Coffin-Manson acceleration.

For high temperature and/or high humidity, Arrhenius and/or Peck equation usually could be used as shown in Eq. 7.

$$\text{AF} = \left(\frac{RH_{\text{low}}}{RH_{\text{high}}}\right)^{n} \times \text{e}^{\left(\frac{E_{\text{a}}}{k_{\text{B}}}\right)\left(\frac{1}{T_{\text{low}}} - \frac{1}{T_{\text{high}}}\right)} \tag{7}$$

The common parameters for Arrhenius and Peck models are listed in Table 8 for reference.

For random vibration, the Basquin model usually could be used to scale the vibration testing time versus G level as shown in Eq. 8.

Table 7 Referenced parameters for Norris-Landzberg model

	AF	Parameter	Lead-free solder	Leaded solder
1	Coffin-Manson	a	2.65	2.5
2	Dwell time	b	0.136	0.0667
3	Ramp rate	c	1.22	0.80094
		d	−0.0757	0.0964
4	Highest temperature	f	2185	1414

Table 8 Referenced parameters for Arrhenius-Peck model

	Parameter	Value
1	n (humidity exponent)	−2.66
2	E_a (activation energy) (eV)	0.8 (average conservative value)
3	k (Boltzmann's constant) (eV/K)	8.6173×10^{-5}

$$G_{\text{RMS - accelerated}} = G_{\text{RMS - normal}} \times \left(\frac{T_{\text{normal}}}{T_{\text{accelerated}}} \right)^{m/2} \tag{8}$$

where m is the Basquin's exponent or fatigue parameter. Some reference numbers for different materials are listed in Table 9.

Table 10 illustrates an example of environmental and mechanical reliability design validation (DV) testing plan for a Compute located in the trunk or vehicle's rear compartment. Total five water-fall legs covering 22 testing items are required for Compute engineering and design validations. Leg 0 includes vibration transmissibility demonstration and thermal cycle profile development, temperature measurement, visual inspection, and design review based on test results (DRBTR), and cross-section (x-section). Leg 1 includes low-temperature wakeup, high-temperature degradation, pothole shock, and random vibration with temperature cycling. Leg 2 includes low-temperature wakeup, non-operational thermal shock or temperature cycling, power temperature cycling (PTC), humidity heat cyclic (HHC), humidity heat constant (HHCO), and salt mist. Leg 3 includes low-temperature wakeup, minimum temperature non-operation temperature storage, dust ingression (IP5k), and water ingression (IP2). Leg 4 includes low-temperature wakeup, collision shock, elbow load, and fretting corrosion for connectors. Among all the tests listed in Leg 1–4 in Table 10, high-temperature degradation, mechanical shock, random vibration, TS, and PTC are considered stress tests or quantitative accelerated life tests. Their acceleration factors can be calculated based on industry-accepted models as described above. The rest tests are considered as performance indicator tests or qualitative accelerated tests. For stress tests, the test duration and sample sizes can be calculated based on Eqs. 4–8 and customized mission profiles per each stress test. For performance indicator tests, the testing durations and sample sizes are recommended to follow the GMW3172 standard.

Table 9 Referenced m values for Basquin vibration model

	Materials	m – Material fatigue constant
1	Aluminum leads in electronic assemblies	6.4
2	Overall usage value	5
3	Connector fatigue or fretting corrosion	4
4	Highly accelerated vibration for metal fatigue (>3.3x original stress)	3.3

Table 10 Compute environmental and mechanical stress tests

	Test A	Test B	Test C	Test D
Test Leg				
DV Leg 0	VTD and TCPD	Temperature measurement	Visual Inspection and DRBTR	X-section
DV Leg 1 water-fall	Low temperature wakeup	High temperature degradation	Shock—Pothole	Vibe w/TC
DV Leg 2 water-fall	Low temperature wakeup	Thermal shock	PTC	Humidity heat cyclic humidity heat constant salt mist
DV Leg 3 water-fall	Low temperature wakeup	Min non-op temperature	Dust (IP5k)	Water (IP2)
DV Leg 4 water-fall	Low temperature wakeup	Shock collision	Elbow Load	Fretting corrosion

It should be noted that a 5-point check before and after each leg is required for all the legs, and a 1-point check at the end of Tests A, B, and C is required per each leg listed in Table 10 except Leg 0.

Compute may be exposed to a variety of different fluids. Exposure to these fluids may affect the functionality of the Compute. The chemical load tests or fluid compatibility tests are intended to assure that vehicle operating liquids, chemicals and oils will not degrade the materials, identification, or function of the Compute. Although other fluids beyond those in the list in Table 11 could come into contact with the Compute, these fluids were considered more aggressive. The following list of fluids in Table 11 was selected based on the likelihood of exposure and the severity of exposure for Compute using liquid coolant located in the trunk or vehicle's rear compartment.

Table 11 Compute chemical load list

Fluid/Chemical/Substance	Specification/Part number	Method
Commercial vehicle cleaning agent-interior	Genuine GM Fluid 88,861,405, Leather, Vinyl and plastic cleaner, Formula 409, Fantastik multi-purpose cleaner, Sonax car interior cleaner	Normal cleaning
Engine Coolant	Ethylene glycol (EG) base fluids, 50:50%	Pour test
Grease, Electrical connector, Dielectric lubricant	9,985,821	Brush test
Ammonia based cleaner	Windex, Sonax glass clear, Glass cleaner	Normal cleaning
Coca-Cola classic		Pour test
Coffee (10 oz., 0.5 oz. Cream, 2 tsp. Sugar)		Pour test

4.3 EMC/ESD Validation

AV Compute EMC testing is the process of measuring the electromagnetic compatibility of a Compute and its components. Automotive EMC testing is now more important than ever as AV component RF design grows in complexity. The main purpose of Compute EMC testing is to test the mutual influence between the Compute and the surrounding electromagnetic environment, which includes the ability of the Compute to resist a given electromagnetic disturbance and the indicators of the electromagnetic disturbance generated by the Compute. That is, the Compute is not affected by the electromagnetic disturbance emitted by other equipment in the electromagnetic environment, and the Compute cannot generate electromagnetic disturbance that exceeds the prescribed limit. Electro Static Discharge (ESD) testing of Compute refers to the transfer of unbalanced charges on the surface of the Compute. When the charge voltage difference is higher than a certain level, the insulating medium will undergo an electrical breakdown process, which will cause a localized conductive path to form inside the insulating medium. Such localized conductive path can induce high current passing through. The main destructive force of electrostatic discharge is the thermal effect from the instantaneous peak current, which can easily cause the Compute electronic components to be broken down or burned, and then cause malfunction of the entire Compute. For safety concerns in automotive electronics, automotive ESD compliance standards have higher voltage test limits than commercial electronics.

The internationally accepted automotive EMC regulations include ECE R10 regulated by the United Nations Economic Commission for Europe (ECE), 97/24/EEC and 95/54/EEC regulated by the European Union, and CISPR (French: Comité International Spécial des Perturbations Radioélectriques), Society of Automotive Engineers (SAE), Japanese Automobile Standards Organization (JASO) and ISO. Generally speaking, EMC/EMI is tested according to customer requirements and specifications in the state of the whole system. Different vehicle OEMs have their own EMC testing specifications such as GMW3091 and 3097 from GM, ES-XW7T-1A278-AC from Ford, TSC3351 from Toyota, DC-10614 and DC10615 from Daimler Chrysler, etc. In this section, we will use GMW3097 as our EMC validation baseline. For EMC testing laboratories, the major U.S. automakers have requested that EMC testing of all components must be performed in a laboratory accredited by Automotive EMC Laboratory Recognition Program (AEMCLRP).

A Compute EMC Test plan should be developed outlining the following per IEC 11451-2:2015:

1. test setup
2. frequency range
3. the reference point(s) (or line if a four-probe method is used)
4. vehicle mode of operation
5. vehicle acceptance criteria
6. definition of test severity levels
7. vehicle monitoring conditions

Table 12 Compute test mode descriptions

DUT test mode number	Operation and description	Function	Used for GMW 3097:2019 procedures
0	Unpowered	Off	ESD handling
1	Full operation	On—Continuous high utilization processing cycle running with traffic on ethernet ports	Emissions tests, Radiated immunity tests, ESD power-up mode
2	Ethernet traffic only	On—Low utilization of processors, full ethernet traffic	Conducted immunity tests

8. modulation
9. polarization
10. Compute orientation
11. antenna location
12. test report content.

For Compute on EV, three modes of operation such as unpowered, ethernet traffic only, and full operation full load should be tested as shown in Table 12.

The full Compute shall be tested in the below set of tests covering both EMC and ESD as shown in Table 13. A generic Compute test setup and a test configuration to be used for all RE tests are shown in Fig. 29. The test should be conducted twice, once with a grounded enclosure and once with an un-grounded enclosure. For isolation, the Compute shall be placed on a non-conductive, low relative permittivity material ($\varepsilon_r \leq 1.4$), at (50 ± 5) mm above the reference ground plane. During the test, all Compute shall not exceed the limits defined by Radiated Emissions Absorber-Lined Chamber (ALSE) Non-Spark Requirements in GMW3097:2019, by Conducted Emissions Artificial Network (AN) Non-Spark Requirements in GMW3097:2019, the "Level 2" requirement for all frequencies and modulations. If Compute passes the component level EMC tests but does not pass the vehicle level EMC tests, the vehicle level test results will be the determining factor for validation test pass/fail status.

ESD test shall verify the immunity of lines, pins, or Compute enclosure locations, which are to be subjected to ESD discharge events. ESD test shall identify the potential ESD discharge points and list all individual pins, case discharge locations, discharge type, simulator voltages, discharge network type, and a description of the pin signal.

Table 14 defines Compute ESD testing for Power-On Mode Setup per GMW 3097:2019 3.6.1 as an example. The test should be conducted twice, once with a grounded enclosure and once with an un-grounded enclosure. For isolation, the Compute shall be placed on a non-conductive, low relative permittivity material ($\varepsilon_r \leq 1.4$), at (50 ± 5) mm above the reference ground plane. During the test, for the test mode(s) given in Table 13, all Compute locations must comply with the performance standards defined in GMW3097:2019. After the test, no permanent

Table 13 Summary of EMC/ESD tests

GMW3097 section	Description
3.3.1	Radiated Emissions—Absorber-Lined Shielded Enclosure
3.3.2	Radio Frequency Conducted Emissions (via Artificial Network)
3.4.1	RF Immunity—Bulk Current Injection
3.4.2	RF Immunity—Anechoic Chamber
3.5.2	Transients Conducted Immunity, Nominal 12 V Lines
3.5.3	CI, Fast Transient Coupling
3.5.4	CI, 30 V DCC Transient Coupling
3.6.1	Electrostatic Discharge, Power on Mode
3.6.2	Electrostatic Discharge, Remote I/O
3.6.3	Electrostatic Discharge, Handling of Devices

Compute damage or performance deviations shall be observed. The Compute ESD power-on test configuration is shown in Fig. 30.

- The DUT is inaccessible from the outside of the vehicle
- Capacitance = 150 pF
- Resistance = 2 kΩ.

Table 15 defines Compute ESD testing for Remote I/O—Inputs/Outputs Setup per GMW 3097:2019 3.6.2 as an example. Remote I/O testing is to be completed on pins 1–8 of each of the two RJ45 Ethernet Service port connectors as well as both PDB LIN lines as shown in Fig. 31. During the test, for the test mode(s) given in Table 15, all Compute pins must comply with the performance standards defined in GMW3097:2019. After the test, no permanent Compute damage or performance deviations shall be observed. This includes changes in rising edge shape in pre/post serial bus plots.

For 4–15 kV.

- Capacitance = 150 pF
- Resistance = 2 kΩ
- Human Body Model (HBM) = 330 pF/2 kΩ for $\leq$ 15 kV; 150 pF/2 kΩ for > 15 kV, unless otherwise specified by GM EMC Engineer.

Table 16 defines Compute ESD testing for Handling of Devices Setup per GMW 3097:2019 3.6.3 as an example. Remote handling testing will be performed on all contactable ports. Both Contact and Air Discharge methods should be attempted. During the test, for the test mode(s) given in Table 16, all Compute ports must comply with the performance standards defined in GMW3097:2019. After the test, no permanent Compute damage or performance deviations shall be observed after exposure.

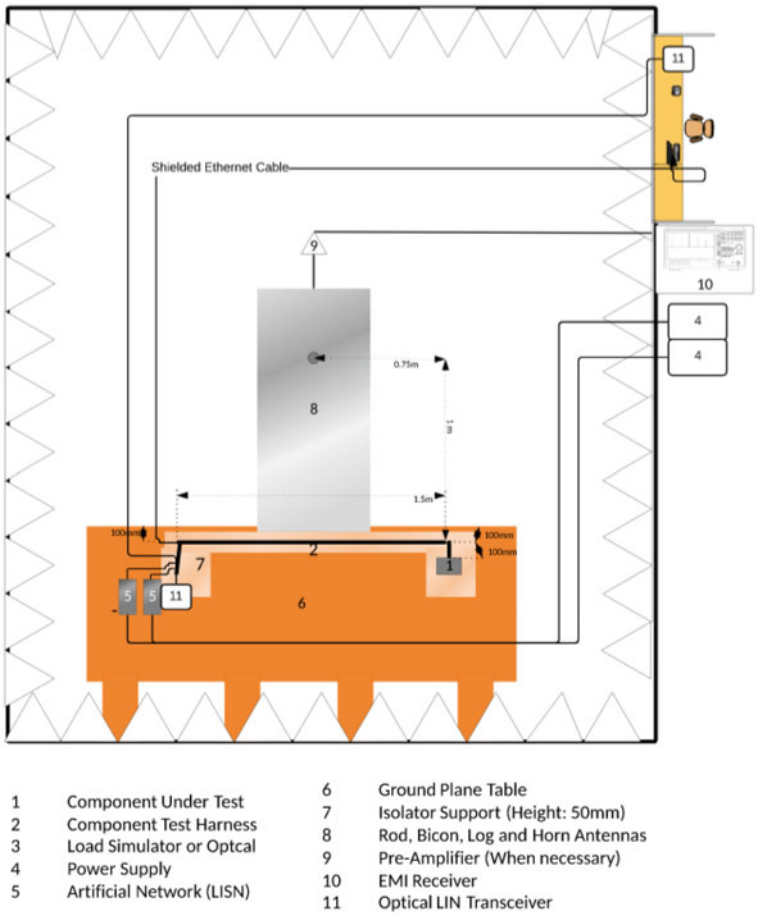

Fig. 29 RE configuration show monopole antenna as a reference, follow CISPR25 for other antennas location

- Capacitance = 150 pF
- Resistance = 2 kΩ.

5 Challenges to Safe Deployment at Scale

5.1 Artificial Intelligence: Perception and Prediction

The perception and prediction rely on the sensors on the autonomous vehicle such as cameras, LiDARs, and radars. Their performance is different in different scenarios. For example, the resolution of 2D image from the camera becomes low in the dark.

Table 14 ESD, test during operation of the device (power-on mode) test

Mode	Location (Pin/Case)	Discharge type (Air/Contact)	ESD simulator voltage (kV)	Signal/Pin description
1	Enclosure surface points	Air	±4	Screw holes/Enclosure edge
1	Enclosure surface points	Contact	±4	Screw holes/Enclosure edge
1	Enclosure surface points	Air	±6	Screw holes/Enclosure edge
1	Enclosure surface points	Contact	±6	Screw holes/Enclosure edge
1	Enclosure surface points	Air	±8	Screw holes/Enclosure edge
1	Enclosure surface points	Contact	±8	Screw holes/Enclosure edge
1	Enclosure surface points	Air	±15	Screw holes/Enclosure edge

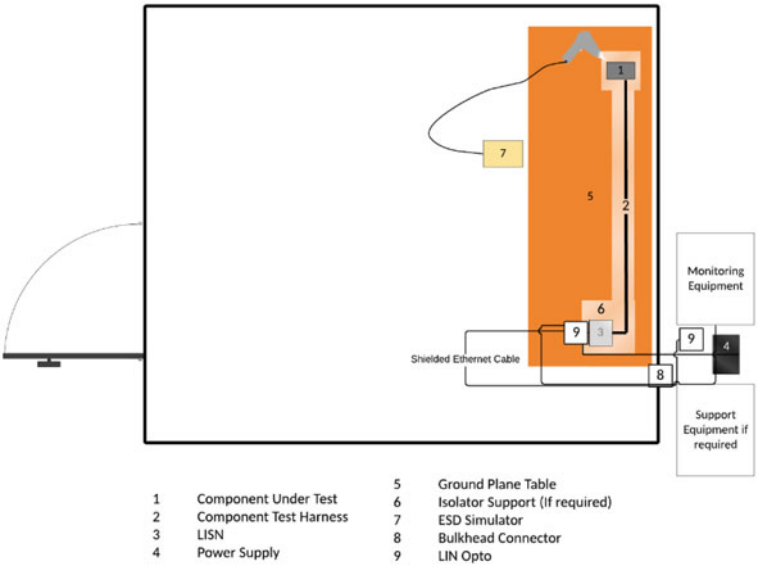

Fig. 30 ESD power-on configuration

Table 15 ESD, remote inputs/outputs test

Mode	Location (Pin/Case)	Discharge type (Air/Contact)	ESD simulator voltage (kV)	Signal/Pin Description/Name
1	Ethernet pins 1–8 on both service ports and both LIN lines	Contact	±4	Ethernet cable for service RJ45 connector and LIN lines
1	Ethernet pins 1–8 on both service ports and both LIN lines	Contact	±6	Ethernet cable for service RJ45 connector and LIN lines
1	Ethernet pins 1–8 on both service ports and both LIN lines	Contact	±8	Ethernet cable for service RJ45 connector and LIN lines
1	Ethernet pins 1–8 on both service ports and both LIN lines	Air	±15	Ethernet cable for service RJ45 connector and LIN lines

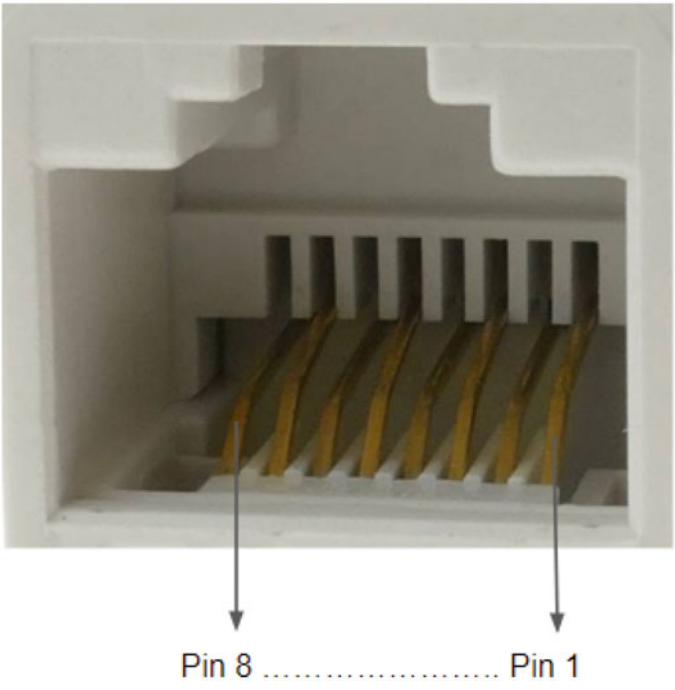

Fig. 31 RJ45 Ethernet port pins

Table 16 ESD, handling of devices test

Mode	Location (Pin/Case)	Discharge type (Air/Contact)	ESD simulator voltage (kV)	Signal/Pin Description/Name
1	All ports	Contact	±4	All ports
1	All ports	Contact	±6	All ports
1	All ports	Air	±8	All ports

On a rainy day or foggy day, the sensor performance will be low. It will impact the perception of an AV. It is challenging for an AV to operate in complex urban streets such as busy intersections in the urban street. Many pedestrians and vehicles appear to be moving in different directions. It is difficult for an autonomous vehicle to do perception, prediction, and make decisions.

5.2 Power Consumption

With government policies for carbon emissions and environmental protection, more and more autonomous vehicles are BEVs. The electric vehicle battery range becomes very impotent. How to increase EV maximum range with autonomous L4 and L5 driving is a big challenge. One option is to increase the battery range. Another option is to reduce autonomous vehicle power consumption.To achieve fully autonomous driving, the autonomous computing platform needs more performance. More performance means more power consumption. For example, the Nvidia Drive AGX is 300 W with 320 TOPS performance. The Tesla D1 Dojo is 400 W with 362TOPS performance. As every watt matters, it is required to design Compute with EVs in mind. One way to do this is to improve efficiencies in the system themselves by designing Compute from the ground up with the EV power platform in mind. It is imperative to have a custom-designed, high density, functionally safe chip, but with lower power consumption to give AV maximum miles on the road. As an example, a new application-specific integrated circuit (ASIC) can achieve more performance but less power consumption as shown in Table 17.

Table 17 Power, performance, and TOPS per watt comparisons of different ASIC chips

ASIC	Power consumption (W)	Performance (TOPS)	TOPS/W
Mobileye Eye Q5	10	24	2.4
Google TPU v3	40	420	10.5
Qualcomm Snapdragon Ride L4/L5	130	700	5.38

5.3 Thermal Management

The autonomous computing platform could generate tremendous heat that increases the component operating temperatures above their temperature limit. Such overheating prevents the components from functioning efficiently, safely, accurately, and reliably. It is critical to control the temperature below the maximum operating temperature limits to prevent them from degrading and malfunctioning.

As long as the autonomous computing platform has large power consumption, it becomes a challenge to cool the temperature below its operating temperature limit. If passive cooling is not able to reduce the temperature, active cooling should be required. Cooling the temperature below the maximum operating temperature is needed to ensure the performance of a Compute. Generally, liquid cooling is used in the autonomous computing platform.

5.4 Manufacturing, Assembly, and Quality Control

After the design phase of the autonomous driving system, it is going to the manufacture and assembly phase. Generally, the manufacturing includes PCB process, PCBA process, function test, enclosure assembly, burin in test, package, and shipping as shown in Fig. 32.

The PCBA process (Fig. 33), includes solder paste printing, place components, reflow soldering, automated optical inspection (AOI), in-circuit test (ICT), image programming, and function test. After that, it is going to the enclosure assembly.

The autonomous driving system is going through a lot of process steps during manufacturing and assembly. It is important how to do quality control and make sure the system has no issues during each process step. Especially, in mass production, how to improve the yield rate becomes challenging.

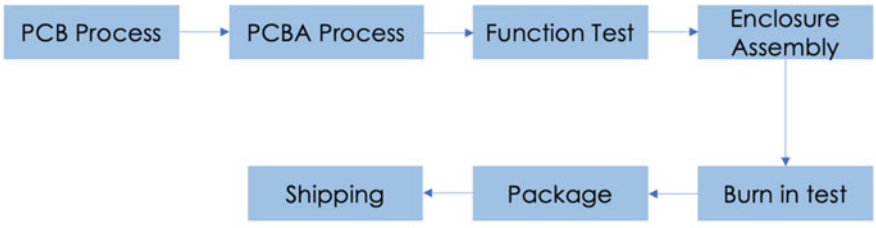

Fig. 32 Autonomous computing system manufacturing process

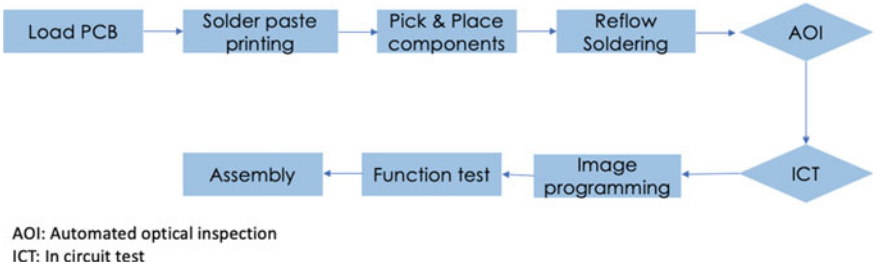

Fig. 33 Autonomous computing system PCBA process

5.5 Size and Cost

For fully autonomous driving, to achieve the performance for perception and prediction, the computing platform needs to use several CPUs, GPUs, and memory to meet the performance requirement. The board and enclosure sizes become larger. Considering the redundancy to make the computing platform safe, several boards are needed in the encloser unit. It will increase the total cost of an AD system.

To reduce the CPU/GPU temperature below the maximum operating temperature, the liquid cooling system is generally used. The enclosure is designed to have heat pipes as well as liquid pipelines or channels. All of them will add to the overall cost of the autonomous driving system.

5.6 Quality and Reliability

The strict reliability standards for AV Compute are critical for road safety and human safety. Since AV Compute would run software with more than 1 billion lines of code during life, hardware reliability is an absolute necessity because a blue screen of a system crush at 60 MPH could mean actual death. Currently, there is no industry well-established reliability target for AV electronic modules such as Compute. Most OEMs and tier-1s adopt automotive industry established traditional vehicle reliability specifications such as 99% or 95% reliability at end of vehicle life to qualify AV electronics. The risk classification scheme of Automotive Safety Integrity Level D (ASIL-D) defined by ISO 26262 places a more stringent reliability standard on self-driving vehicles. For these vehicles to be ASIL-D compliant, the maximum acceptable probabilistic metric for random hardware failure (PMHF) is 10FIT. In other words, these vehicles can only make ten errors in 1 billion hours of operation, while an average U.S. driver makes 10,000 mistakes in the same duration. As an example, for a Compute to achieve a 10FIT failure rate at the end of 5 years of life with an 80% duty factor, the reliability target will be 99.965% instead of 99 or 95%, which is a great challenge.

Failures in computer systems are broadly categorized into permanent hard failure and intermittent recoverable soft faults. Permanent hard failures are repeatable and occur the same way every time. On the contrary, intermittent soft faults are temporary and are a function of the operating environment and stress loading. While permanent hard failures sound scary, they are relatively easier to handle in general. A diligent reliability testing framework usually can expose permanent hard failures, therefore they can be mitigated by design and process optimization. In the worst case, they can be monitored, diagnosed, and quarantined by on-vehicle safety measures. But intermittent faults are often harder to be diagnosed so to be prevented since they are a function of the unique operating environment and stress loading. How to address intermitted recoverable faults is another great challenge for Compute validation and usage.

As mentioned in an early section, AV Compute's operation time and mileage mission profiles could be 2–3 times of the traditional human-driving vehicles. With such longer daily continuous operation hours or mileages, the reliability specifications for AV hardware especially Compute shall be higher, therefore it will be challenging. Furthermore, with such long continuous operation, the probability of a vehicle hitting extreme road conditions or corner cases increases drastically. For environmental loads such as thermal, mechanical, radiation, dust, water, humidity, chemical, etc., we can't just use traditional values such as 95th or 98th, or even 99th percentiles to model such loads for the use conditions. We may have to adopt the absolute worst case from a 5- or 10-year period to truly guard band Compute's durability. In addition to bad environmental conditions, poor infrastructure and chaotic road conditions are also proving to be tremendously challenging for Compute operation.

In the worst case, a Compute with redundant GPUs and CPUs could consume more than 2000 W of power. Therefore, an enormous amount of heat would be generated. As autonomous driving functions rely heavily on the Compute power of the data processing units, the clock speeds must stay in the optimal range all the time. Therefore, air cooling usually is not adequate to meet the thermal management requirement. Instead, liquid cooling offers a much higher cooling capacity and is therefore generally chosen as the Compute thermal solution. Using a liquid cooling coldplate to enclose a Compute will impose several new reliability challenges. First, the coldplate usually serves as a good "moat" to insulate the temperatures of boards/components inside from external temperature changes as shown in Fig. 34. Therefore, the traditional reliability testing methods described by GMW3172 do not apply to liquid cooling Compute. The boards and components inside the coldplate will see liquid-to-liquid induced temperature changes instead of air-to-air induced temperature changes. Since a liquid is used as the thermal medium, very high thermal ramp rates can be achieved with liquid-to-liquid temperature change, when compared to the air-to-air temperature change. As a result, the liquid-to-liquid temperature change is considered more stringent stress than air-to-air in terms of acceleration. As a consequence, a more severe board and coldplate interaction would be generated. Second, for a liquid cooling Compute, during its validation testing and field operation/maintenance, condensation effect, hydrolocked state effect, and water hammer effect all need to be diligently investigated and assessed. As illustrated in Fig. 35,

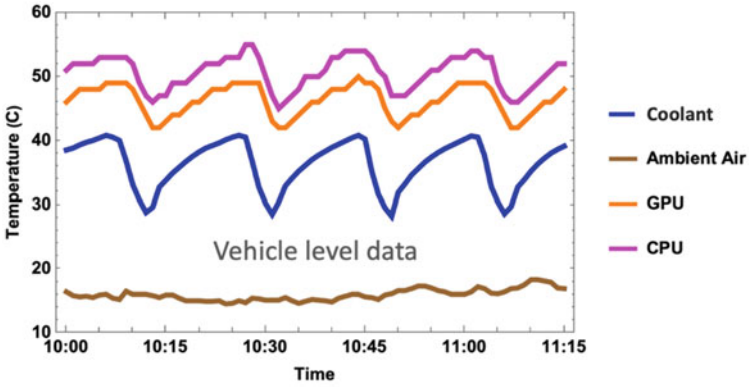

Fig. 34 Compute GPU and CPU temperatures follow coolant closely, while they are independent of ambient air temperature

a hydrolocked state-induced hydrostatic pressure under a high-temperature stress condition can deform the coldplate severely to cause Compute failure. Third, special unit handling, chiller pump operating, and transporting procedures also need to be carefully developed. As an example, at the end of Compute testing, wait until coolant temperature reaches room temperature before unplugging the Compute connector. To prevent a water hammer, first, unplug the connector of the water inlet, then unplug the connector of the water outlet.

For a Compute to be automotive qualified, all the ICs and electronic components used on the board need to meet AECQ standards throughout the manufacturing and testing process first. AECQ is a set of failure mechanism-based stress test qualifications. Among them, AEC-Q100 is for packaged integrated circuits, AEC-Q101 is for active components, AEC-Q102 is for optoelectronic devices, AEC-Q200 is for passive components, and AEC-Q104 is for multi-chip module (MCM) used in automotive applications. This specification has been established by the Automotive Electronics Council (AEC) to define qualification requirements and procedures for ICs and electronic components used in the automotive industry. An AEC-qualified device means that the device has passed the specified stress tests and guarantees a certain level of quality and reliability. Unfortunately, currently on the market, there has been none AEC-qualified CPUs ever available. Furthermore, due to supply chain shortage issues and some other reasons, it is not common that non-automotive-grade ICs and components have to be used for Compute. Such usages bring a great challenge for Compute for its long-term reliability and defect-free quality requirement for harsh environment operation. In general, a thorough gap analysis with component and board level validations is needed to assess the real risks if non-automotive-grade ICs and components are going to be used for AV Compute.

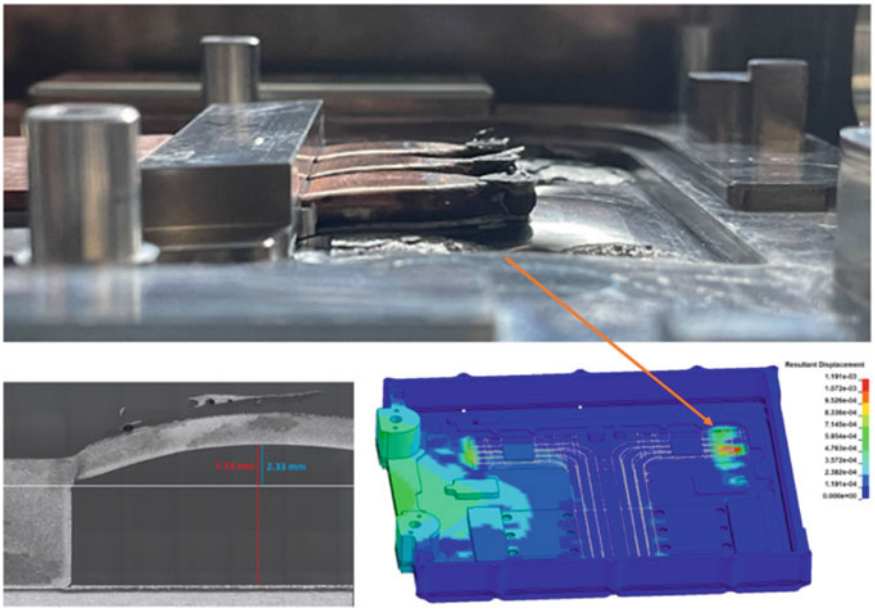

Fig. 35 Compute failure caused by coldplate buckling due to liquid thermal expansion under a hydrolocked state. Maximum in-plane stress of 134,121 psi was generated that caused buckling of 2.57 mm based on simulation, in good agreement with the actual cross-section measurement

Reliability has sometimes been classified as "how quality changes over time". Building Compute to achieve high reliability requires setting and achieving standards for precise process and assembly. Keeping Compute stresses within the design envelope during operation requires setting and meeting precise operating domain that delivers the least-stress operating performance. These standards are called quality standards. For AV Compute, if we want it to be highly reliable, we must first set the appropriate process and operating quality standards. Then we must achieve those standards. Hence, a high level of quality assurance is required to deliver its matching reliability. Setting and achieving world-class quality standards would bring world-class reliability. Higher reliability then brings higher safety. For a component to be automotive qualified, manufacturers have to meet specific industry standards throughout the manufacturing and testing process. IATF 16949 is a global automotive industry standard for such quality management and control. The automotive industry generally expects parts to be manufactured, assembled, and tested in IATF 16949 qualified facilities. However, currently, not all AV component suppliers and contract manufacturers are IAFT 16949 certified. The AV companies who are not traditional vehicle OEMs are also likely not IAFT 16949 certified. Furthermore, there is still a question on if IAFT 16949 is adequate for building AV hardware. AV's high reliability and safety standards require a matured supply chain with a higher level of the quality management system. The current IATF 16949 quality management system focuses mainly on the quality part, not being adapted to effectively

include the security activities and safety aspects. This could be the main weakness of the current IATF 16949. Therefore, there still is a great challenge to integrate quality, security, and safety standards to synthesize a coherent quality management system for the development of AV Compute.

5.7 Security and Safety

Each AV is equipped with or supported by Compute to process the sensor data, monitor the vehicle's status, and control the mechanical components. Hence, the security threats against the Compute are of serious concern. Specifically, the attacks targeting AVs could cause fatal traffic accidents, and threaten both personal and public safety. There are many methods for AV attacks. How to defend against these attacks to ensure the safety and security of AV is of paramount challenge.

The safety of AV is at risk if security is compromised at any level. As each AV is equipped with numerous sensors and a Compute, an attacker targets one of the sensors, the Compute, or the communication networks to confuse, mislead, or even take over control of the vehicle under attack, leading to fatal accidents. It is extremely dangerous for any AVs to go on the road if it fails to meet the safety and security requirements. Generally speaking, it is extremely difficult to enter the Compute system. However, the vehicle infotainment system and the OBD-II port of the overhaul system are all connected to the CAN bus, and the CAN bus is connected to Compute, which allows hackers to enter Compute. The methods of attack include the following:

- Onboard diagnostics (OBD)-II intrusion: the OBD-II port is mainly used to diagnose the status of the vehicle, firmware update, and vehicle control. Usually, when the vehicle is in service, the technician will use the detection software developed by each car OEM to access the OBD-II port and exam the vehicle. Since OBD-II is connected to the CAN bus, as long as hackers obtain such detection software, they can easily hack into the vehicle system.
- Attack the AV remote control management platform: car schedule and resource allocation are all controlled by this cloud platform Therefore, once the platform is attacked by hackers, the entire AV dispatch and control system of a city may be disrupted, and traffic paralysis and accidents are prone to occur.
- Invasion of electric vehicle chargers: with electric vehicles becoming more and more popular, charging equipment has become an indispensable core component of the electric vehicle ecosystem. Since the EV charging unit will communicate with an external charging station during charging, and the charging unit will be connected to the CAN bus, this allows hackers to invade the CAN system through the external charging station.
- Car media player intrusion: there has been an attack case where the attack code is encoded into the burned music CD [22]. When the user plays the CD, the

malicious attack code will invade the CAN bus through the CD player, to obtain bus control and steal the core information of the vehicle.

- USB invasion: USB and other input and output interfaces. Plug a special USB into the car's USB port to complete certain car functions. If a USB is compromised with built-in chips, ROM, RAM, and wireless network functions, as well as written malicious control programs. If the line connection and signal transmission are large enough, and the Compute and other important ECU modules are involved, the safety of the AV and the safety of information can be damaged.
- Bluetooth intrusion: another entry point for attacks is Bluetooth. Nowadays, Bluetooth connection of mobile phones and car communication and entertainment systems has become standard. Since users can send and read information to and from CAN via Bluetooth, this also gives hackers a window to attack. In addition to gaining control of the owner's mobile phone, because the effective range of Bluetooth is 10 m, hackers can also use Bluetooth to carry out remote attacks.
- TPMS invasion: TPMS is a wheel pressure management system. Hackers can also launch attacks on TPMS. In this attack method, the hacker first places the attack code in the vehicle TPMS ECU, and then when the TPMS detects a certain tire pressure value, the malicious code will be activated to attack the vehicle.

A general solution is to encrypt and verify the information received by the Compute to ensure that the information is sent by a trusted MCU or component, not by a hacker. Using encrypted authentication, symmetric or asymmetric ciphers can be chosen. The symmetric cipher has a small amount of calculation, but the two parties in the communication need to know the cipher in advance. The asymmetric key does not require the password to be known in advance, but it is computationally intensive. Such additional safety authentication and encryption may cause Compute processing latency and communication timeout to impact AV operation. Therefore, it is necessary to consider increasing the delay caused by the safety mechanism while verifying the safety. Finally, the distribution and management of ciphers are also crucial but challenging. Although in recent years there have been some interesting proposals regarding protecting the security of AVs, more research is required before we deploy AVs on a large scale.

6 Summary

More and more fatalities associated with early developed AVs arise recently, which reveals the big gap between the current AV Compute system and the expected robust system for L4 and L5 full AD. In this chapter, we gave a high-level review of computing systems for autonomous driving, including an ADAS overview, centralized Compute system architecture, functional test and validation, and challenges. Safety and reliability are the most important requirements for autonomous vehicles.

Hence, the challenge of designing a Compute ecosystem for AVs is to deliver enough computing power, redundancy, and security to guarantee the safety and reliability of AVs while consuming less power.

Acknowledgements The authors want to thank Dr. Daniel Braun from BMW Group for his critical reviews of this chapter.

References

1. *World Health Organization*, [online] Available: https://www.who.int/news/item/11-12-2010-pedestrians-cyclists-among-main-road-traffic-crash-victims
2. *International Organization for Standardization (ISO)*, [online] Available: https://www.iso.org/standard/43464.html
3. Brian Krzanich. Data is the New Oil in the Future of Automated Driving. Intel Newsroom. 2016.11.15
4. Charles Murray. What's the Best Computing Architecture for the Autonomous Car? Design-News. 2017.08.17
5. Apollo Auto, https://github.com/ApolloAuto/apollo
6. https://apollo.auto/platform/hardware.html
7. (2018). *Meet NVIDIA Xavier: A New Brain for Self-Driving, AI, and AR Cars.* [Online]. Available: https://www.slashgear.com/meet-nvidia-xavier-a-new-brain-for-self-driving-ai-and-ar-cars-07513987/
8. (2020). *Enabling Next Generation ADAS and AD Systems.* [Online]. Available: https://www.xilinx.com/products/silicon-devices/soc/xa-zynq-ultrascale-mpsoc.html
9. *Texas Instruments TDA*. https://www.ti.com/lit/wp/spry272a/spry272a.pdf
10. (2020). *The Evolution of EyeQ.* [Online]. Available: https://www.mobileye.com/our-technology/evolution-eyeq-chip/
11. Cong Hao, et. al., 2019 IEEE International Workshop on Signal Processing Systems, pg 121–126, 2019
12. [Online]. Available: https://cloud.google.com/blog/products/gcp/an-in-depth-look-at-googles-first-tensor-processing-unit-tpu
13. https://www.youtube.com/watch?v=j0z4FweCy4M&t=6750s
14. L. Liu, S. Lu, R. Zhong, B. Wu, Y. Yao, Q. Zhang, and W. Shi, IEEE INTERNET OF THINGS JOURNAL, VOL. 8, NO. 8, APRIL 15, 2021
15. (2017) Moving from 24 GHz to 77 GHz radar. [Online]. Available: https://www.edn.com/moving-from-24-ghz-to-77-ghz-radar/
16. S. Liu , L. Liu, J. Tang, B. Yu, Y. Wang, and W. Shi. Edge Computing for Autonomous Driving: Opportunities and Challenges
17. System on a chip. [Online]. Available: https://en.wikipedia.org/wiki/System_on_a_chip
18. x86. [Online]. Available: https://en.wikipedia.org/wiki/X86
19. Ultimate Guide to Real-time Operating Systems (RTOS). [Online]. Available: https://blackberry.qnx.com/en/rtos/what-is-real-time-operating-system/
20. Hualiang, et. al., ECTC 2021–1559
21. Endo, Tatsuo; Mitsunaga, Koichi; Takahashi, Kiyohum; Kobayashi, Kakuichi; Matsuishi, Masanori (1974). "Damage evaluation of metals for random or varying loading—three aspects of rain flow method". *Mechanical Behavior of Materials.* **1**: 371–380.
22. S. Checkoway et al., "Comprehensive experimental analyses of automotive attack surfaces," in Proc. USENIX Secur. Symp., San Francisco, CA, SA, 2011, pp. 77–92.