A 360-degree view of surround-view and automated parking systems

Texas Instruments

John Smrstik
Product Marketing
Jacinto™ Processors

Aish Dubey
Systems Engineer
Jacinto Processors

Texas Instruments

Introduction

Arriving home late at night to a garage filled with bicycles, sports equipment and yard gear, you hope there is enough room to fit your car inside for the night.

On a rainy and cold morning, the office parking lot is nearly full, with every available spot seemingly miles from the front door.

Heading into town on a Friday night, the only thing between you and the start of a fun evening is determining whether or not your car will fit into the only available parking spot—and still let you squeeze out of the vehicle.

These are typical scenarios that can be frustrating for drivers who are in a hurry, trying to avoid the elements, or just seeking a little convenience. Fortunately, automated driving capabilities will ease these frustrations and make the driving experience more convenient and enjoyable, even for mundane tasks like parking a car.

Basic surround-view systems give drivers visual cues to make them more informed and aware of their surroundings. Deep learning applied to the video images captured by the car's cameras enable more sophisticated services, such as locating open parking spaces, automating the parking process and enabling driverless "automated valet" capabilities.

These automated capabilities (including multiple camera inputs, visual perception and scene creation) require powerful processing in the automobile. Processors for advanced driver assistance system (ADAS) applications need to be able to combine megabytes of visual or other sensor data to produce an interpreted environment for the car—an environment reliable enough to safely operate the vehicle at low speeds with or without a driver. Texas Instruments (TI) designed its Jacinto™ 7 processor family exactly for this challenge.

In this white paper, we will explain how automotive customers can build ADAS applications with TI Jacinto TDA4VM devices for assisted and fully automated parking functions. We will map the technical requirements for such systems across all segments of the automotive market (compact, midsize, large and luxury) to the Jacinto TDA4VM silicon device and software platform, and explain how you can introduce technology needed for safe and comfortable fully automated parking.

Automated parking and parking assist systems

It is possible to separate parking assist systems into three basic classes based on the capabilities of the system. **Table 1** presents a higher-level view of the classes.

Basic surround-view systems use multiple camera inputs to give drivers a 360-degree view of the area immediately around the car. Camera inputs are stitched into a single bird's-eye view, with the car at the center. The image is presented to the driver as visual information to assist in manual parking. Overlays that show car position relative to objects, curbs or painted parking lines enhance the surround-view image.

System type	Description of capabilities	Number and type of sensors used	Required algorithms	Required device features
Surround-view systems (aka surround-view monitoring)	Gives drivers a 360-degree 2D or 3D view of the area surrounding the vehicle to assist with manual parking	• Four to six cameras	• 2D and 3D view creation based on the graphics processing unit (GPU) and digital signal processor (DSP) • Sometimes, specialized hardware accelerators can be used for limited 3D view generation	• Embedded ISP • GPU or HWA for view creation
Semi-automated self-parking	Driver identifies an available space. The car takes over the parking task into parallel or perpendicular spaces. The system is also able to automatically "un-park". The driver remains in ultimate control of the system.	• Four to six cameras • Six to 12 sonar sensors • Four to six short-range radar sensors • Inertial measurement units (IMUs) • Localization hardware and software	• Parking spot and lane detection • Near- and far-field object detection and classification at low speeds • Simultaneous localization and mapping (SLAM) • L4 level of path planning for a given map and driving conditions • Encoding and recording of video streams to log vehicle actions	• Embedded ISP • Embedded HWA for visual analytics and convolutional neural network (CNN) processing • ASIL-D processing for limited sensor fusion and decision communication • GPU • MCU (ASIL-B) DMIPS for general-purpose, limited-performance path planning and sensor fusion
Fully automated valet parking	The car is able to autonomously park and un-park itself in a well-defined parking region or parking lot. The car will identify available spots. The driver will not be in control of the vehicle during any part of this process.	• Four to six cameras • Six to 12 sonar sensors • Four to six short-range radars • IMUs • Localization hardware and software	• Parking spot and lane detection • Near- and far-field object detection and classification at low speeds • SLAM • Encoding and recording of video streams to log vehicle actions	• ASIL-D SLAM • Embedded ISP • Embedded HWA for visual analytics, path planning and CNN processing • ASIL-D processing for full-scope sensor fusion • ASIL-D DMIPS for path planning • On-chip video encoder

***Table 1**. Surround-view and automated parking applications and requirements.*

The next class of parking assist systems are semi-automatic self-parking systems. These systems combine camera, ultrasound and positional information to form a more informed picture of the environment around the car, and thus enable some level of parking automation. Based on this information, the car will take over basic parking tasks to allow automated maneuvering into (or out of) available parallel or perpendicular parking spaces by controlling steering, brakes, acceleration and gear shifting. In these scenarios, the driver first identifies an open space and remains in full control of the system to take over automated parking tasks as needed.

Fully automated valet parking systems take the next step, allowing the car to fully park and unpark itself in a well-defined parking area. The driver cedes control of the vehicle during the parking process, starting with the identification of an available space. This application requires more sensor input and more sophisticated processing and algorithms in order to reliably and safely execute parking tasks on its own.

Each scenario—starting from basic surround view to fully automated valet parking—requires an increasing number of sensors, data and data processing. A processor system on chip (SoC) built to enable these applications needs:

- an image input pipeline processing,
- general-purpose processing,
- specific acceleration for deep-learning tasks,

- graphics processing for the creation of images with overlays and
- Automotive Safety Integrity Level (ASIL)-rated processing to help assure safe operation of the systems.

Table 1 specifies the algorithms and on-chip features needed for each system class. The absolute performance required for each of these features—such as deep learning tera-operations per second (DLTOPS), Dhrystone million instructions per second (DMIPS), billions of floating-point operations per second (GFLOPS) or the megapixel processing capability of an image signal processor (ISP) or hardware accelerator (HWA) engine—may further separate system types into smaller subsegments.

How the Jacinto family of TDA4VM processors handles surround-view and automated parking challenges

As you may have gathered from the information in **Table 1**, there are several challenges ahead for system architects and business teams at automakers and Tier 1 suppliers tasked with introducing these features in production automobiles. First, automakers want to offer a range of features across their models—simple surround features on economy vehicles, with growing levels of insight and autonomy on midsize and luxury cars. Each model has varying economic realities: economy models cannot include the same type of electronics as high-end luxury models. Changing processor platforms to develop and validate new software for each model type is time-consuming and expensive.

Tier 1 suppliers prefer a platform that offers a common approach for implementing solutions that can scale from low- to high-end vehicles by adding additional sensors and cameras to the base design. Reusing hardware and software assets realizes the necessary engineering efficiencies to minimize R&D expenses and get to market quickly with different product alternatives.

The Jacinto TDA4VM processor family and TI's processor software development kit (SDK) combined give OEMs and Tier 1 suppliers a new way to solve this problem. These devices feature heterogeneous processing to deliver application performance while managing power consumption to enable use in embedded spaces with thermal and size constraints. TI's Jacinto TDA4VM SoC uses hardware acceleration, custom processor cores, signal processors, general-purpose processors and microcontrollers (MCUs) to help designers create an efficient system solution. TI selected and designed each intellectual property (IP) component to solve specific problems and to be adaptable across a range of end system requirements.

Table 2 describes the processing steps and IP components typically used in simple surround-view monitoring applications, as well as more complex automated valet parking cases.

Processing step	SoC IP component
Image capture	Imaging input/output (Camera Serial Interface [CSI]-2)
Image processing	ISP and vision acceleration components
Analytic processing for feature extraction	DSP and deep learning accelerator
Fusion processing (from multisensor inputs)	General-purpose processor, MCU
Display image preparation	GPU
Display output	Multidisplay subsystem

***Table 2**. Processing steps in surround-view and automated parking applications, and the SoC IP used.*

The heterogeneous approach requires specific software for each type of processor core or accelerator. It is possible to abstract low-level software stacks, optimized for the cores, using higher-level software concepts to help simplify development and to provide access to the high-performance cores. OpenVX is one such software framework; it is open, royalty-free and designed for real-time embedded vision processing. The TI processor SDK uses OpenVX-based examples that show how to build applications (like surround-view monitoring) leveraging the software components in the SDK.

Putting it all together: The Jacinto TDA4VM SoC

The basic application requirements for parking assist and automation applications require specific functions for acquiring camera and image sensor data, and preparing that data for the processing stage. The processing stage takes the image data and performs analytics and deep learning algorithms to extract key features relevant to parking applications. This stage integrates (or fuses) data from other sensors to form a more complete environment around the car, and presents a picture to the application for decision-making—in this case, safely maneuvering the car into and out of a parking space. A final step is the presentation of image data to the driver in an intuitive manner to aid them in safely operating the vehicle. Preserving the video data for possible review in the future is also a necessary step, particularly in fully automated situations. All of these steps have to take place in a functional safety environment that provides redundancy and logically (or physically) separates mission-critical functions from other operations.

TI considered all of these application requirements in the design of the TDA4VM SoC. Based on an understanding of the system and with the goal to provide solutions that are efficient, flexible and easy to use, the TDA4VM SoC incorporates components to handle capture, processing and rendering requirements. One of the principal design choices entailed balancing the processing and data requirements in order to ensure enough local memory and proper access to high-speed external memory, while keeping the processing systems operating at high efficiency.

Figure 1 shows a simplified block diagram of the TDA4VM device in a surround-view use case

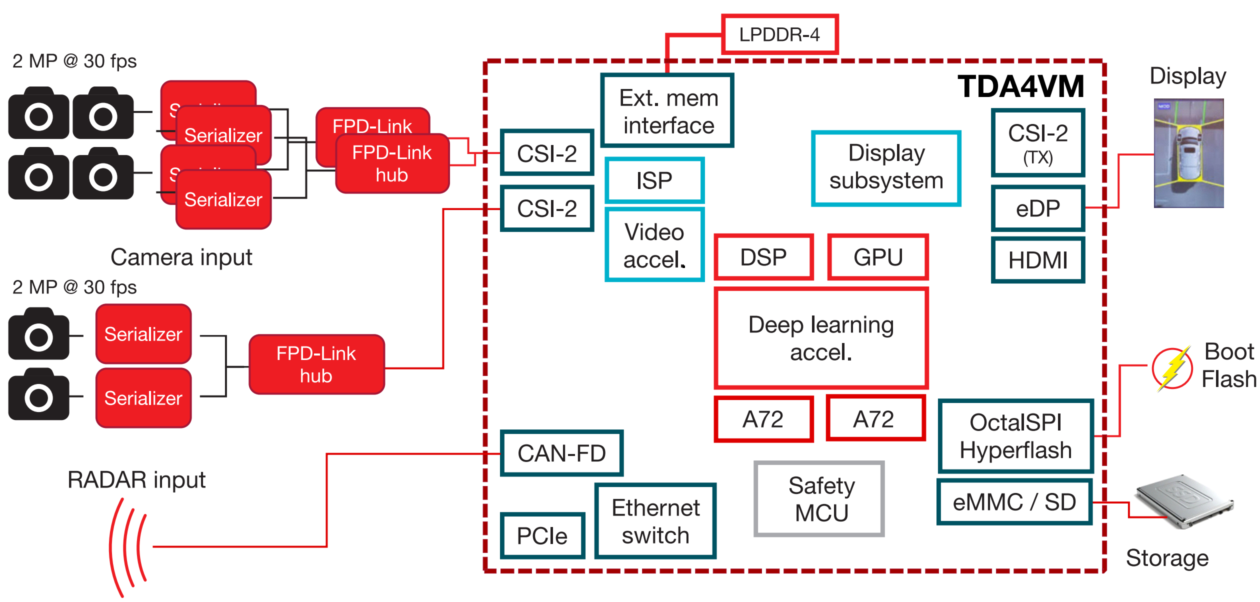

***Figure 1**. Simplified surround-view system based on TDA4VM.*

Surround-view/automated parking application stage	TDA4VM SoC features
Image data capture	• CSI-2 interface for low-voltage differential signaling cameras • Ethernet for Ethernet-based cameras • Seventh-generation TI ISP • Lens distortion correction hardware accelerator • Noise filter hardware accelerator • Multiscalar hardware accelerator
Sensor fusion and analytics processing	• C66x and C7x DSP cores • Tightly coupled matrix multiply accelerator for deep learning applications • Dense optical flow hardware accelerator • Stereo depth engine hardware accelerator • Two Arm® Cortex®-A72 central processing units • Four Arm Cortex-R5F MCUs • Controller Area Network-Flexible Data Rate and Universal Asynchronous Receiver Transmitter interfaces • Large internal shared memory (8 MB) • High-efficiency, high-bandwidth on-chip fabric • Low power-double data rate 4 with 17-GBps peak bandwidth
Image display and preservation	• Imagination GE8430 GPU core (100 GFLOPS) • H.264 encode hardware • H.265 decode hardware • 4K and 2.5K resolution video display output
Safety and security	• Dual-lock-step R5F MCU (ASIL-C safety island) • System approach to IP security and protection from intrusion and attacks

***Table 3**. Surround view application stages mapped to TDA4VM device features.*

showing video and other sensor input, display output and access to storage for compressed video files. **Table 3** describes the processing stages in a surround-view and automated parking application, and the key features of the TDA4VM device that support those processing stages. As described, the TDA4VM device is a very complete "system on chip" for these parking applications.

No chip solution is complete without a software environment to accompany the chip. The TDA4VM SoC is supported by complete software kits for Linux and the TI real time operating system (RTOS) kernel. Called the processor SDK, these kits contain a full set of drivers, operating system kernels, application libraries, boot examples, OpenVX-based application frameworks and application examples that show how to use the software and hardware components in real system applications. These kits are validated on the evaluation module for the devices that are available from TI.

The Jacinto TDA4 is a planned family of devices, of which the TDA4VM is just the first. Other members of this processor family will include various combinations of the same chip-level IP to provide more optimized products for all corners of the ADAS market, whether those are analytic-heavy applications or more cost-optimized needs for consumer-class vehicles. Because these devices are built from the same basic hardware IP and software technology, compatibility among the devices remains intact. Software assets developed to accommodate one device can scale and be reused on other devices in the family, streamlining development efficiency and making it easier to introduce a complete product line with varying feature sets across vehicle models. Surround-view systems moving toward park assist and automated valet parking are one collection of applications that you can easily develop using Jacinto family technology.

Additional resources

- Learn more about **Jacinto TDA4x processors** for ADAS applications.
- Get more information on **TI solutions and design resources** for ADAS.
- Explore our portfolio of **FPD-Link camera serializers and deserializers** for ADAS applications

IMPORTANT NOTICE AND DISCLAIMER