Computer Architectures for Vision-Based Advanced Driver Assistance Systems

Merlando *******, Harry Broeders
*******, 1376934
Computer Architecture (Special Topics), ET4078
merlando@*********, J.Z.M.Broeders@student.tudelft.nl

Abstract

High-end cars are equipped with all kind of systems to assist the driver. These systems provide more comfort and safety. A large part of these systems is vision-based. Three application specific processors for such systems are examined in this article. The processors are compared and based on predictions on the future of advanced driver assistance systems suggestions for improvements are given. In particular the applicability of a reconfigurable architecture is considered. It is concluded that two of the examined application specific processors, the EyeQ2 and VIP-II, can be enhanced by using a reconfigurable architecture. The third processor, IMAP-CAR, can gain more from the addition of general purpose processor cores. How much gain there is in using a reconfigurable architecture for vision-based advanced driver assistance system is still open for research.

1 Introduction

Lack of attention to relevant driving events is one of the main factors in traffic accidents [7]. Recently so called advanced driver assistance systems are build into commercial cars to assist the driver and prevent accidents from happening. A large part of these systems is vision-based. This paper gives an overview of application specific computer architectures used in vision-based advanced driver assistance systems. In section 2 currently available advanced driver assistance systems are discussed. In section 3 the requirements for such systems are given. Section 4 presents some application specific processors which are able to fulfill there requirements. These processors are compared in section 5. In section 6 future trends are identified and discussed. In particular the applicability of a reconfigurable architecture, like the MOLEN processor [12], is discussed in section 7.

2 Background

At the moment (mid 2008) there is a large range of advanced driver assistance systems available in commercially available cars. A recent overview can be found in section 5.1.1 of [5]. For example:

- Adaptive Cruise Control.
  The cruise control slows down when you close in on the vehicle in front of you.
- Collision Warning and Auto Break.
- Pre-crash Safety.
  Tightens seatbelts, puts chairs in best position and puts neck rests up just before the crash.
- Lane Departure Warning.
- Lane Keeping Assistant.
  Actively keeps the car into its own lane.
- Stop and Go Assistant.
  Assists traffic-jam driving.
- Blind Spot Detection / Lane Change Assistant.
- Night vision.

- Parking Assistant.
- Driver Drowsiness Detection.
  Monitors the eyelid position of the driver and warns when he/she doze off.
- Traffic Sign Recognition.
- Pedestrian Detection.

Most of the benefits of advanced driver assistance systems are obvious: more safety and comfort for the driver and other traffic participants. But there are also some less obvious benefits, most of which are due to the fact that the assisted driver moves more smoothly through traffic: better utilization of the infrastructure, lower energy consumption and less pollution [1]. There are also some disadvantages. Users of advanced driver assistance systems have the tendency to perform other activities, because they have a false sense of security. Some active assistants can give the driver an improper feedback. The line keep assistant, for example, can create a feeling of cross wind. In all current systems the driver is still ”in the loop”. This is an advantage, because the driver can intervene when the system makes a mistake, but also a disadvantage, because the driver can make a wrong intervention.

In current advanced driver assistance systems different types of sensors are used [10]. RADAR (RAdio Detection And Ranging), ultrasonic, LIDAR (LIght Detection And Ranging) and camera based systems are in use. The first 3 systems in the previous list are active systems; they all transmit a signal and receive the reflection of this signal. This works fine in isolation but when all vehicles in a crowded environment transmit for example a radar signal, all these signals are going to interfere. Solutions for this interference problem are not trivial. The camera system on the other hand is a passive system, it only receives information.

There are two kinds of camera based vision systems used at the moment: monocular and stereo vision. Stereo vision has the disadvantage that it, obviously, needs a second camera and that it needs calibrations. Most current vision-based advanced driver assistance systems are therefore monocular at the moment. Due to the fact that digital cameras are mass produced products they are far cheaper than radar and lidar systems. Besides these advantages monocular vision systems have some disadvantages. The system is dependable on lighting and weather conditions. The system must be able to assist the driver in bright daylight, at night, when driving in a tunnel, during rainfall, in a snowstorm etc. An other disadvantage for lane recognition systems is that several different colors are used for temporary road markings in Europe (yellow, red or blue).

The image processing task that an advanced driver assistance system needs to perform can be divided into low level and high level tasks. The low level tasks are: filtering, edge, corner and line detection and thinning operations. These low level tasks can employ pixel level parallelism. The high level tasks are: detection, classification and tracking (of lanes, vehicles, pedestrians, traffic signs, driver’s eyelids, etc), motion estimation, 3D modeling etc. These high level tasks can utilize task level parallelism.

An example of the image processing needed in an advanced driver assistance system is lane recognition [13]. This task can be split in the following steps: edge enhancement, edge detection, corner detection, line marker detection and, finally, lane recognition. The results of the lane detection algorithm can be used to measure the distance to the vehicle in front of you and by comparing adjacent frames the speed relative to the vehicle in front can be determined.

3 Requirements

Computer systems used in advanced driver assistance system need to fulfill the usual requirements of the automotive industry. Such as low cost, low power usage, high production quality, high temperature range (-40 °C ... +85 °C), high reliability and high accuracy. These requirements all originate from the fact that these computer systems are meant to be applied in mass produced cars and perform live critical functions.

Specific requirements for computer systems used in advanced driver assistance system are a high flexibility, small size and much computational power. These computer systems must be flexible due to the wide variety of applications that potentially can run concurrently on these systems. At the moment most of the available advanced driver assistance systems only performs one function. But in the future adaptive cruise control, collision warning and auto break, pre-crash safety, lane keeping assistant, stop and go assistant, night vision, traffic sign recognition and pedestrian detection can all use the same camera and computer system. The camera generates much data, therefore the computer system which has to process this data and the camera are often positioned close together. Because the camera needs to be placed in a position where not much space is available, for example after the rearview mirror, the computer system used in an advanced driver assistance system needs to be small. All advanced driver assistance systems must fulfill several specific temporal requirements. Therefore these systems can be qualified as real-time systems. The large amount of sensory data that must be processed in real-time and the large number of applications, which potentially have to run on the system, causes an excessive demand of processing power.

4 Application Specific Processors

In this paper three application specific processors for advanced driver assistance systems are discussed: IMAPCAR [3], EyeQ2 [8] and VIP-II [11]. The IMAPCAR is produced by NEC [4] since 2006 and is applied for example by Toyota Lexus in their cars [6]. The EyeQ2 is developed by Mobileye and STMicroelectronics [9] and is scheduled for production in 2009. It is the successor of the EyeQ1 that is applied for example in cars from BMW and Volvo. The VIP-II is the successor of the VIP-I and is developed by Infineon. We could not find a car manufacturer that applies this processor.

4.1 IMAPCAR

The IMAPCAR (Integrated Memory Array Processor for CARs) is a SIMD linear array processor, see figure 1. It contains 128 8-bit 4-way VLIW RISC processing elements each with a 24-bit multiply-accumulate unit. Each processing element has 2K of local data memory and can perform 4 instructions in one clock cycle. The processor elements are interconnected via a shift-register style ring network. A single 16-bit RISC control processor with 32KB program and 2KB data caches is used to control the 128 processing elements.

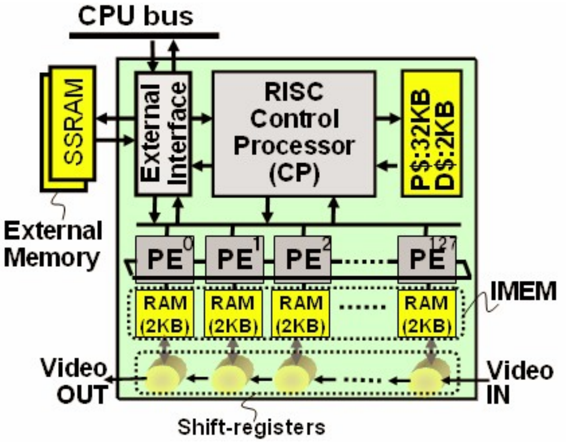

Figure 1: IMAPCAR organisation

The IMAPCAR is programmed in a C language extension specially developed for this architecture, which is called 1DC (one dimensional C). This language supports the usual vector operations such as vector arithmetic, data-dependent operations, indirect table look-up, selective operations and status collection. An image is loaded column wise into the 128 local memories. A processing element has therefore direct access to all pixels in a column of the picture. Using the ring network each processing element can perform a direct register-to-register data access of neighborhood processing elements. This enables the IMAPCAR to fully explore the pixel level parallelism found in many low level image processing tasks. One shortcoming of the IMAPCAR design is that it is not easy to exploit the task level parallelism found in high level image processing tasks.

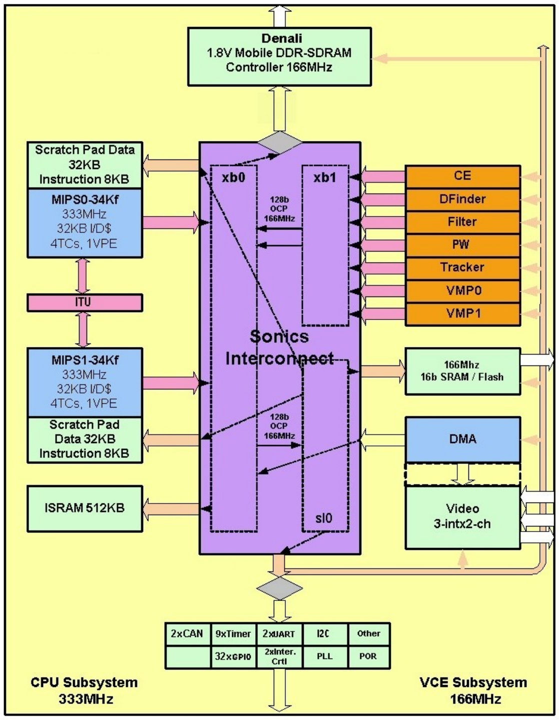

Figure 2: EyeQ2 organisation

4.2 EyeQ2

The EyeQ2, which is shown in figure 2, contains 2 MIPS32 34Kf processor cores. These cores have hardware support for multi-threading. So called thread contexts are provided which consists of a PC and a register file. Each thread can be assigned to a virtual processing element. Each virtual processing element has its own translation look aside buffer for the cache and behaves like a stand alone processor. The 34Kf core can support up to 9 thread contexts in two virtual processing elements. The two MIPS cores can quickly exchange data using the ITU (Inter Thread Communication Unit). Besides these 2 MIPS cores the EyeQ2 has 7 so called vision computing engines. These are fixed logic processing elements to address image pre-processing, object classification and object tracking. The vision computing engines available in the EyeQ2 are CE (Classifier Engine), DFinder (Disparity Finder) which is used for stereo vision, Filter, PW (Preprocessor Window), Tracker which is used for motion analysis and 2 Vector Microcode Processors which utilize parallel vector, scalar and table lookup units. The 2 MIPS cores and the 7 vision computing engines are connected with an interconnection network from Sonics called SMX (Sonics Multi-service eXchange). This sophisticated network on chip allows the system to run 7 different vision computing engines simultaneously. The interconnect takes care of the interaction between the engines, and many other issues are solved by the interconnect like interrupt management, data-width conversion, power management and pipelined stages for requests/responds [2].

Information about how to program this architecture is not provided by Mobileye or STMicroelectronics at the moment.

4.3 VIP-II

The VIP-II (Vision Instruction Processor version 2) features 4 multi-tasked SIMD cores. Each core provides 4 processing elements as shown in figure 3. The cores use VLIW instructions to perform arithmetic operations and memory access in parallel. Each processing element is pipelined (4 stages). Data dependencies between the pipeline stages are completely avoided because each of the pipeline stages operates on an instruction from another thread. To make this possible each processing element is provided with 4 instruction caches, 4 register files and 4 program counters. Every SIMD core is controlled by a general purpose core. The four cores are connected via a multi-layer system bus. An additional general purpose processor (ARM9) handles the communication tasks and main control flow.

The 4 general purpose controllers within the SIMD cores and the ARM9 are all programmed in C. To program the SIMD cores a C language extension, called DPCE (Data Parallel C Extension), is used. This extension supports, as the name suggests, data parallel programming.

5 Comparison

When the 3 architectures discussed above are compared a lot of commonalities can be found.

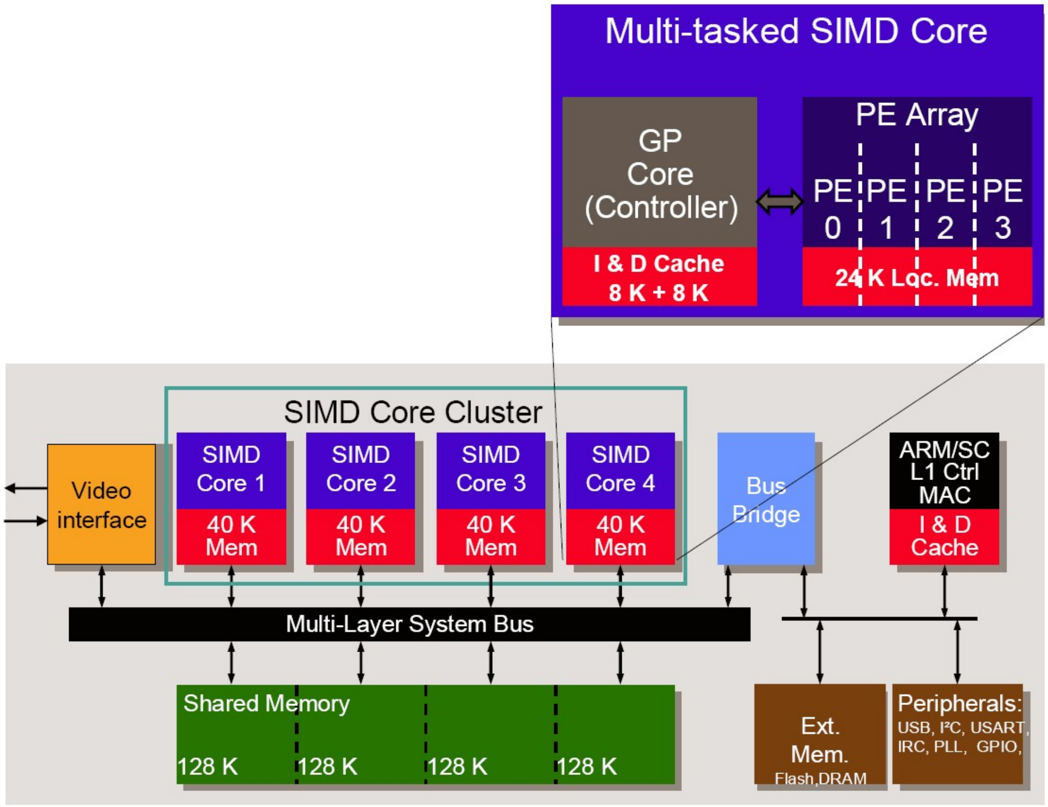

Figure 3: VIP-II organisation

All chips run at relatively low clock frequencies (a few 100 MHz) and have low power consumption (a few watts). All processors support SIMD instructions to explore the pixel level parallelism found in low level image processing tasks. All chips have a video interface to input the video signal from the camera and to output a video signal for display purposes, for example in a night vision system.

The following differences can be observed. The IMAPCAR and VIP-II both keep the video data close to the processing elements (in local memories) but the EyeQ2 utilizes a network on chip from Sonics to access the video data from the different processing elements. The EyeQ2 uses a DMA controller for data-transfers between the Sonics interconnect and the video input/output. Both the IMAPCAR and VIP-II provide their own C language extension to program the chip, the program language used for the EyeQ2 is not known. The IMAPCAR has no special support for MIMD parallelism to explore the task level parallelism found in high level image processing tasks. The EyeQ2 and VIP-II both have multiple cores which provide hardware support for multi-threading to exploit task level parallelism. The IMAPCAR is in production for 2 years now while the EyeQ2 and VIP-II are only prototypes. The EyeQ2 is the only processor that provides processing elements which are dedicated to a specific task. The IMAPCAR is a typical vector processor with a SIMD architecture. There is only one controller which controls the 128 identical processing elements. The EyeQ2 is mainly a MIMD architecture but it also contains two Vector Microcode Processors. The name suggests that these have a SIMD architecture but detailed information about them is not available. The VIP-II has 16 processing elements; each group of four is controlled by one controller. Such a group is called a SIMD core and, obviously, has a SIMD architecture. But because there are 4 SIMD cores working in parallel this processor also has a MIMD architecture.

6 Future Trends

Some future trends in advanced driver assistance systems can be predicted. Vision-based systems will be merged with radar, ladar and/or ultrasonic based systems. Each sensor type has specific advantages. For example, radar has a longer range than vision and is less sensitive to weather conditions, but radar can not be used to detect pedestrians and road signs. Therefore information coming from different types of sensors will complement each other.

At the moment advanced driver assistance systems work in isolation, but in the future the advanced driver systems of cars which are in each others neighborhood will work together using vehicle to vehicle communication. Using this feature a car ahead, which is just around the corner, can warn a car which comes behind for an obstacle it can not observe itself.

Currently the traffic in crowded places is constantly monitored and regulated by traffic monitor and control systems. Communication between advanced driver assistance systems and traffic monitor and control systems can further enhance the functionality of both systems. The traffic monitor system can, for example, warn the advanced driver assistance system for a traffic jam ahead. The advanced driver assistance system can, for example, inform the traffic monitor system of an emergency break.

The application specific processors of future advanced driver assistance systems will have to process more data, will have to run more concurrent tasks and will have to do a lot more communicating.

7 Reconfigurable Processors

As just explained, application specific processors for advanced driver assistance systems will need a huge amount of computational power. One of the possibilities to provide this power is the use of reconfigurable processors such as the MOLEN processor [12]. For low level image processing various pre-processing algorithms are required according to weather and lightning conditions. Because these algorithms are very computational intensive, a dedicated piece of hardware can be used to perform these algorithms. When the weather or lightning conditions change the dedicated hardware can be reconfigured to perform another, more effective, algorithm. Also some applications are only needed at certain moments in time or in certain situations. Night vision is, obviously, only needed when it is dark and pedestrian recognition is not needed when driving on the highway. When there are computational intensive operations used in these applications then a hardware module which can perform these operations can be configured when needed and reconfigured when no longer needed.

From the application specific architectures discussed in this paper the EyeQ2 architecture is the only one which contains dedicated hard-wired processing elements. The EyeQ2 architecture can therefore simply be enhanced by making these processing elements reconfigurable. For example the DFinder (Disparity Finder) processing element which is used for stereo vision can then be reconfigured to perform some other function in a monocular vision system, where there is no need for the DFinder. A reconfigurable variant of the EyeQ2 can contain some MIPS cores and some reconfigurable processing elements all connected with the SMX interconnection network.

How the VIP-II and IMAPCAR can benefit from reconfigurable processing elements is less clear. The IMAPCAR lacks the possibility to explore task level parallelism at the moment; therefore this architecture will probably gain more from the addition of some general purpose processor cores then from the addition of reconfigurable processing elements. The VIP-II can be extended with one or more SIMD cores which are not controlled by a general purpose core but which perform a reconfigurable hard-wired algorithm. These reconfigurable SIMD core can then be used to execute the most computational intensive operations.

Of course further research is needed to investigate how much application specific processors, used in advanced driver assistance systems, can gain by using reconfigurable processing elements.

8 Summary and Conclusion

Recently a lot of vision-based advanced driver assistance systems became available to the drivers of cars from the high-end of the automotive market. Application specific processors used in current and future systems include the IMAPCAR, EyeQ2 and VIP-II. The applications which run on advanced driver assistance systems contain both pixel level parallelism and task level parallelism. The application specific processors which run these applications use SIMD processing, MIMD processing, vector processing, VLIW instructions and hardware multi-threading to exploit this parallelism. All investigated processors have a high-speed connection to the video-data, which allows the systems to perform real-time pixel-level operations. The IMAPCAR has the best support for pixel level parallelism but no support for task level parallelism at all. The EyeQ2 and VIP-II have a more balanced design and provide support for both kinds of parallelism by merging SIMD and MIMD processing.

Future systems will be integrated with other systems and these future systems will require more computational power. The addition of reconfigurable processing elements can help to meet this requirement for the EyeQ2 processor and probably also for the VIP-II processor. The IMAPCAR processor will probably gain more from the addition of some general purpose processor cores. More research is needed to find out how much can be gained from the addition of reconfigurable processing elements to application specific processors for vision-based advanced driver assistance systems.

References

[1] Tom Alkim, Gerben Bootsma, and Pierre Looman. The assisted driver – systems that support driving. Technical report, Ministry of Transport, Public Works and Water Management, Rijkswaterstaat, 2007.

[2] Jeff Haight. Advanced interconnects drive intelligent vision applications part 1 and 2. *Automotive DesignLine*, July and September 2007.

[3] Shorin Kyo and Shin'ichiro Okazaki. In-vehicle vision processors for driver assistance systems. In *Proc. Asia and South Pacific Design Automation Conference ASPDAC 2008*, pages 383–388, 21–24 March 2008.

[4] NEC. Nec electronics and nec introduce imapcar image processor with advanced parallel processing capabilities for automotive safety systems new processor adopted in lexus ls460 precrash safety system. Press release, August 2006.

[5] Johan Norén. Warning systems design in a glass cockpit environment. Master's thesis, Linköping University, Department of Management and Engineering, 2008.

[6] Motoyuki Ooishi. New toyota lexus detects pedestrians, applies brakes. *Nikkei Electronics Asia*, November 2006.

[7] K. Rumar. The basic driver error: Late detection. *Ergonomics*, 33:1281–1290, 1990.

[8] Elchanan Rushinek and Pete Del Vecchio. Multi-threaded design tackles soc performance bottlenecks. *EmbeddedSystems Europe*, pages 26–28, Aug 2006.

[9] STMicroelectronics and Mobileye. Stmicroelectronics and mobileye deliver second-generation system-on-chip for vision-based driver assistance systems. Press release, May 2008.

[10] Zehang Sun, G. Bebis, and R. Miller. On-road vehicle detection: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28(5):694–711, May 2006.

[11] Axel Techmer. Application development of camera-based driver assistance systems on a programmable multi-processor architecture. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 1211–1216, 13–15 June 2007.

[12] S. Vassiliadis, S. Wong, G. N. Gaydadjiev, K.L.M. Bertels, G.K. Kuzmanov, and E. Moscu Panainte. The molen polymorphic processor. *IEEE Transactions on Computers*, pages 1363–1375, November 2004.

[13] Joo young Yi, Sang hoon Yang, In gul Jang, Jin gyun Chung, and Myoung seob Lim. Automobile advance alarm system based on monocular vision processing. In *Proc. IEEE Intelligent Vehicles Symposium*, pages 428–432, 13–15 June 2007.