THE DESIGN AND IMPLEMENTATION OF ADVANCED DRIVER ASSISTANCE SYSTEM (ADAS) DATA ACQUISITON ENGINE BASED ON HETEROGENEOUS COMPUTATION PLATFORM

FAWAZ MOHAMMED JUMAAH

SULTAN IDRIS EDUCATION UNIVERSITY

2021

THE DESIGN AND IMPLEMENTATION OF ADVANCED DRIVER ASSISTANCE SYSTEM (ADAS) DATA ACQUISITON ENGINE BASED ON HETEROGENEOUS COMPUTATION PLATFORM

FAWAZ MOHAMMED JUMAAH

05-4506832 pustaka.upsi.edu.my Perpustakaan Tuanku Bainun Kampus Sultan Abdul Jalil Shah PustakaTBainun ptbupsi

THESIS SUBMITTED IN FULFILLMENT OF THE REQUIREMENT FOR THE DEGREE OF DOCTOR OF PHILOSOPHY

FACULTY OF ART, COMPUTING AND CREATIVE INDUSTRY
SULTAN IDRIS EDUCATION UNIVERSITY

2021

05-4506832 pustaka.upsi.edu.my Perpustakaan Tuanku Bainun Kampus Sultan Abdul Jalil Shah PustakaTBainun ptbupsi

UNIVERSITI PENDIDIKAN SULTAN IDRIS
SULTAN IDRIS EDUCATION UNIVERSITY

UPSI/IPS-3/BO 32
Pind: 00 m/s: 1/1

Please tick (✓)

Project Paper	
Masters by Research	
Masters by Mix Mode	
Ph.D.	✓

DECLARATION OF ORIGINAL WORK

This declaration is made on the 13/08/2021.

i. Student's Declaration:

I Fawaz Mohammed Jumaah –P20171001177- Faculty of Art, Computing and Creative Industry hereby declares that the dissertation /thesis for Doctor of Philosophy titled "The Design and Implementation of Advanced Driver Assistance System (ADAS) Based on Heterogeneous Computation Platform" is my original work. I have not plagiarized from any other scholar's work and any sources that contain copyright had been cited properly for the permitted meanings. Any quotations, excerpt, reference, or re-publication from or any works that have copyright had been clearly and well cited.

Signature of the student

ii. Supervisor's Declaration:

I Dr. Bilal Bahaa Zaidan hereby certify that the work entitled, "The Design and Implementation of Advanced Driver Assistance System (ADAS) Based on Heterogeneous Computation Platform" was prepared by the above-named student and was submitted to the Institute of Graduate Studies as a partial / full fulfilment for the conferment of the requirements for Doctor of Philosophy (By Research), and the aforementioned work, to the best of my knowledge, is the said student's work.

23 – August - 2021

Date

DR. BILAL BAHAA ZAIDAN
Faculty of Art Computing and
Creative Industry, UPSI
hp: 010 420 4377
Date: / /20

Signature of the Supervisor

UPSI/IPS-3/BO 31
Pind.: 01 m/s:1/1

INSTITUT PENGAJIAN SISWAZAH /
INSTITUTE OF GRADUATE STUDIES

BORANG PENGESAHAN PENYERAHAN TESIS/DISERTASI/LAPORAN KERTAS PROJEK
DECLARATION OF THESIS/DISSERTATION/PROJECT PAPER FORM

Tajuk / Title: **The Design and Implementation of Advanced Driver Assistance System (ADAS) Based on Heterogeneous Computation Platform**

No. Matrik /Matric's No.: **P20171001177**

Saya / I **Fawaz Mohammed Jumaah**

mengaku membenarkan Tesis/Disertasi/Laporan Kertas Projek (Kedoktoran/Sarjana)* ini disimpan di Universiti Pendidikan Sultan Idris (Perpustakaan Tuanku Bainun) dengan syarat-syarat kegunaan seperti berikut:-

acknowledged that Universiti Pendidikan Sultan Idris (Tuanku Bainun Library) reserves the right as follows:-

1. Tesis/Disertasi/Laporan Kertas Projek ini adalah hak milik UPSI.
   The thesis is the property of Universiti Pendidikan Sultan Idris
2. Perpustakaan Tuanku Bainun dibenarkan membuat salinan untuk tujuan rujukan dan penyelidikan.
   Tuanku Bainun Library has the right to make copies for the purpose of reference and research.
3. Perpustakaan dibenarkan membuat salinan Tesis/Disertasi ini sebagai bahan pertukaran antara Institusi Pengajian Tinggi.
   The Library has the right to make copies of the thesis for academic exchange.
4. Sila tandakan (√) bagi pilihan kategori di bawah / *Please tick (√) for category below:-*

☐	SULIT/CONFIDENTIAL	Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub dalam Akta Rahsia Rasmi 1972. / Contains confidential information under the Official Secret Act 1972
☐	TERHAD/RESTRICTED	Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan ini dijalankan. / Contains restircted information as specified by the organization where research was done.
☑	TIDAK TERHAD / OPEN ACCESS	

(Tandatangan Pelajar/ Signature)

DR. BILAL BAHAA ZAIDAN
Faculty of Art Computing and Creative Industry, UPSI
hp: 010 420 4377
Date: / /20

(Tandatangan Penyelia / Signature of Supervisor) & (Nama & Cop Rasmi / Name & Official Stamp)

Tarikh: 23 – August - 2021

Catatan: Jika Tesis/Disertasi ini SULIT @ TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh laporan ini perlu dikelaskan sebagai SULIT dan TERHAD.

Notes: If the thesis is CONFIDENTAL or RESTRICTED, please attach with the letter from the organization with period and reasons for confidentiality or restriction.

DEDICATION

“And mankind have not been given of knowledge except a little”

Holy Qur’an / Al-Israa (85)

To the memory of my father, whether he died or still alive …

To my mother, the beautiful smile in my life…

To my wife, the soul mate of my heart…

To my daughter, the happiness of my dreams …

To my beloved friends for their encouragement and help

Thank you for being there with me…

ACKNOWLEDGMENT

"In the name of Allah, the most beneficent and the most merciful"

First and foremost, all praise is due to Almighty Allah subhanahu wa-ta'ala, on whom we ultimately depend on, for His endless blessing and allowing me to achieve what I have achieved today.

I would like to take this opportunity to express the deepest thanks and appreciation to the chairman of supervisory committee, Dr. Bilal Bahaa Zaidan for being a great supervisor, motivator as well as teacher throughout the long journey it took to complete this thesis. Furthermore, I would like to thank my supervisory committee members, Associate Professor Dr. Aos Alaa Zaidan and Associate Professor Suzani Binti Mohamad Samuri for serving as my committee members and providing their insightful comments and suggestions.

My special thanks to Intel Corporation for the financial and technical support during my study journey. Special thanks to Melvin The, Hazlina Ramlee and Remesh Seth Nair for their dedicated support and encourage.

My special and hearty gratitude go to my family especially my mother and my lovely wife, for their unreserved love and support. My parents sacrificed their lives since I was child until now to raise me up. Nothing I can say or do that can reward you for what you have done. My beloved wife was beside me throughout my study facing many hard moments. I really appreciate your patience, dedication, and true support.

ABSTRACT

ADAS supports drivers with the required tools and augments to properly decide while driving the car. It can safely control the car either by providing relevant information around the car to the driver, or by taking control of the vehicle movement, partially or completely. The purpose of this study is to develop an ADAS Data Acquisition Engine based on heterogenous platform that utilises SoC and FPGA platforms. Furthermore, the design methodology of heterogenous SoC-FPGA platform is developed to reduce design complexity of heterogenous design flow. Furthermore, it unifies the hardware and software design flows to reduce design cycle time required for development. The proposed system was verified indoor to confirm system functionality. After that, the proposed system was implemented on car for real-time system validation and testing. The system was able to interact with LiDAR sensor, four ultrasonic sensors, and Inertial Movement Unit (IMU) sensors. The LiDAR and ultrasonic were used for long distance, and short distance measurements, respectively. The proposed system implemented on FPGA consumed 15% of logic resources, and 76% of internal memory. The proposed test plan has been derived based on case study reliability tests, system functionality tests, data validation tests, and ADAS application functionality tests. Each test was executed four times to ensure system reliability. The proposed system was able to detect objects in short-range perspective through the ultrasonic sensor from 20 centimetres to 450 centimetres. Furthermore, the system was able to detect long-range distance through the LiDAR from 4 meters, up to 70 meters. The car steering wheel orientation was measured through the IMU sensor ranged from 58.8 angle (clockwise steering movement) to -61.4 angle (anti-clockwise steering movement). The collected data was postprocessed through Rapidminer studio software tool and was presented for further Artificial Intelligence (AI) future applications.

REKA BENTUK DAN PELAKSANAAN SISTEM BANTUAN PEMANDU LUAR BIASA (ADAS) MESIN PENGAMBILAN DATA BERDASARKAN HASIL

ABSTRAK

ADAS menyokong pemandu dengan keperluan alat dan pertambahan dalam membuat keputusan dengan betul semasa memandu kereta. Ia dapat mengawal kereta dengan selamat sama ada dengan memberikan maklumat yang relevan di dalam kereta kepada pemandu, atau dengan mengawal pergerakan kenderaan, samaada sebahagian atau sepenuhnya. Tujuan kajian ini adalah untuk mengembangkan Mesin Perolehan Data ADAS berdasarkan platform heterogen yang menggunakan platform SoC dan FPGA. Selanjutnya, metodologi reka bentuk platform SoC-FPGA heterogen dikembangkan untuk mengurangkan kerumitan reka bentuk aliran reka bentuk heterogen. Selain itu, ia juga menyatukan aliran reka bentuk perkakasan dan perisian untuk mengurangkan masa kitaran reka bentuk yang diperlukan untuk pembangunan. Sistem yang dicadangkan telah disahkan di dalam rumah untuk mengesahkan fungsi sistem. Selepas itu, sistem yang dicadangkan akan dilaksanakan pada kereta untuk pengesahan dan pengujian sistem masa nyata. Sistem ini dapat berinteraksi dengan sensor LiDAR, empat sensor ultrasonik, dan sensor Inertial Movement Unit (IMU). LiDAR dan ultrasonik masing-masing digunakan untuk mengukur jarak jauh, dan jarak pendek. Sistem cadangan yang dilaksanakan pada FPGA menghabiskan 15% sumber logik, dan 76% memori dalaman. Rancangan ujian yang dicadangkan telah dibuat berdasarkan ujian kebolehpercayaan kajian kes, ujian fungsi sistem, ujian pengesahan data, dan ujian fungsi aplikasi ADAS. Setiap ujian dijalankan empat kali untuk memastikan kebolehpercayaan sistem. Sistem yang dicadangkan dapat mengesan objek dalam perspektif jarak pendek melalui sensor ultrasonik dari 20 sentimeter hingga 450 sentimeter. Selanjutnya, sistem ini dapat mengesan jarak jarak jauh melalui LiDAR dari 4 meter, hingga 70 meter. Orientasi roda stereng kereta diukur melalui sensor IMU berkisar dari sudut 58.8 (pergerakan stereng mengikut arah jam) hingga sudut -61.4 (pergerakan stereng berlawanan arah jam). Data yang dikumpulkan diproses pasca melalui alat perisian studio Rapidminer dan disajikan untuk aplikasi Artificial Intelligence (AI) di masa hadapan.

TABLE OF CONTENTS

CHAPTER 2 LITERATURE REVIEW ON ADAS SYSTEMS DESIGN

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ACC	Adaptive Cruise Control
ADAS	Advanced Driver Assistance Systems
AEB	Automatic Emergency Braking
AI	Artificial Intelligence
API	Application Program Interface
ASE	Address Span Extender
ASIC	Application-Specific Integrated Circuit
AXI	Advanced eXtensible Interface
BIST	Built-In Self-Test
BSP	Board Support Package
CMS	Camera-Monitor Systems
CNN	Convolution Neural Network
COTS	Commercial-Off-The-Shelf components
CPM	orientation-adaptive and ComPuting Module
CPU	Central Processing Unit
DAE	Data Acquisition Engine
DARPA	U.S. Defence Advanced Research Projects Agency
EBS	Emergency Braking System
EVE	Embedded Vision Engine
F2H	FPGA-to-HPS
FHD	Full High Definition
FL	Front Left
FLC	Forward Looking Camera

FPGA	Field Programmable Gate Array
FR	Front Right
GPIO	General Purpose In/Out
GPS	Global Positioning System
GPU	Graphics Processing Unit
H2F	HPS-to-FPGA
HAL	Hardware Abstraction Layer
HAS	Heterogenous System Architecture
HDL	Hardware Description Language
HiL	Hardware-in-the-Loop
HOG	Histograms of Oriented Gradients
HPS	Hard Processor System
HT	Hough transform
I2C	Inter-Integrated Circuit
IMU	Inertial Measurement Unit
IP	Intellectual property
ITS	Intelligent Transportation System
JTAG	Joint Test Action Group
LDW	Lane Departure Warning
MCO	Movement Order Control
MPSoC	Multi-Processor System-on-Chip
mSGDMA	modular Scatter-Gather Direct Memory Access
MTT	Multiple Target Tracking
OA-NLM	Orientation-Adaptive Non-Local Mean
OD	Object detection

PIO	Peripheral Input/Output
PLD	Programmable Logic Devices
PLL	Phase Lock Loop
PREM	PRedicable Execution Model
PVP	Pipelined Vision Processor
RAM	Random-Access Memory
RL	Rear Left
ROI	Region Of Interest
Rootfs	Root File System
RR	Rear Right
RTL	Register Transfer Level
SAD	Sum of Absolute Differences
SFM	Structure From Motion
SGM	Semi-Global Matching
SM	Streaming Multiprocessors
SoC	System-on-Chip
sof	Software Object File
SPI	Serial Peripheral Interface
SRDF	Single Rate Data Flow
SSOF	Scene Optical Flow Estimation
SV	Surround View
SVM	Support Vector Machine
TOPS	Tera Operations Per Second
UART	Universal Asynchronous Receiver-Transmitter
V2V	Vehicle to Vehicle

VeHiL	Vehicle Hardware-in-the-Loop
VSC	Vehicle Stability Control
WHO	World Health Organization
XSG	Xilinx System Generator

LIST OF APPENDICES

CHAPTER 1

RESEARCH INTRODUCTION

1.1 Introduction

This chapter outlines the work by providing a brief background on the research, clarifying the problem of research, and then presenting the ambitions and motivation of this research, which is accomplished by achieving the research objectives. Furthermore, it emphasises the research questions and scope.

In next section 1.2, a brief research background regarding the research topic is presented. Section 1.3 defines the research problem statement. In sections 1.4 and 1.5, the research objectives and research questions are presented, respectively. The scope of this study is presented in section 1.6, while the significance of this research is presented

is in section 1.8. Finally, outline the main structure of the thesis are briefly reported in section 1.9. Finally, section 1.10 summarises chapter 1.

1.2 Research Background

The major cause of death worldwide is the traffic accidents. In December 2018, the World Health Organization (WHO) reported that, around 1.35 million people die in traffic accidents every year, and up to 50 million are injured (*World Health Organization Report*, 2020). As a result, autonomous and assisted driving become important to avoid accidents in roads. In 2004, a series of autonomous car races were launched by the U.S. Defence Advanced Research Projects Agency (DARPA). These races offered uniform and modern testing opportunities to examine the autonomous driving (Iagnemma & Buehler, 2006). The technology behind the autonomous driving has been enhanced as a long-term goal for automotive manufacturers, whereas the ADAS has been proposed as the short-term plan to improve road safety gradually.

ADAS has been one of the core fields of innovation in the automotive industry in recent years, starting with assistance systems to aid drivers in critical but infrequently appearing situations. The cost of this evolution can be seen in the increased complexity of algorithms, systems, and sensors (Hiltschera et al., 2018). The recent CMOS technology nodes enabled implementation of a complex multi-processor system on a single chip (MPSoC) that may involve hundreds of different parallel processors, substantial memory, and communication resources, and realize high-performance

computations in an energy efficient way. This facilitated a further rapid progress in autonomous computing, combined with progress in sensors (Jóźwiak, 2017).

As of today, systems supporting these huge computational loads are extremely power-hungry, making them practically impossible to commercialize. The converging needs for predictable high-performance at low power call for a "real-time embedded super-computing platform", i.e., a platform capable of predictably providing real-time guaran- tees to applications running on top of power-efficient embedded hardware. Modern Heterogeneous architectures based on multi- and many-core accelerators can satisfy this need for energy-efficient performance (Burgio et al., 2017). Integrating multiple computing elements running at lower frequencies allows obtaining impressive performance capabilities at a reduced power consumption, while architectural heterogeneity enhances platform flexibility (Burgio et al., 2017). Towards this end, there are three elements need to be addressed, namely, ADAS, Field Programmable Gate Array (FPGA), and Heterogenous System Architecture (HAS). The following sub-sections describe the mentioned elements.

1.2.1 Advanced Driving Assisted Systems (ADAS)

ADAS is a technology that supports a driver in the complex processes of safely controlling a vehicle either by simply providing the driver with relevant information about the surrounding environment or by taking partial control of the vehicles movement to prevent potential accident (Hammond et al., 2015). The automotive

industry continues to look at ways to reduce the fatalities and the severity of road accidents. To achieve this, various innovations have been proposed in the past decades, such as automatic airbag deployment, antilock braking systems, lane-departure warning systems, etc. However, the ever-rising number of cars plying the roads necessitates additional improvements and additions to the existing control systems to reduce road accidents and resulting loss of life and property (Zhong et al., 2016).

The last few years have seen the introduction of more sophisticated safety systems, which are collectively termed ADAS, that integrate sensors such as radar or camera to track objects, such as other vehicles, cyclists, or pedestrians, around a moving car to better gauge and maintain a safe distance at all times (J. Zhang et al., 2011). Numerous ADAS functions have been developed to help drivers avoid accidents, improve driving efficiency, and reduce driver fatigue (Vahidi & Eskandarian, 2003). Lane Departure Warning (LDW) is important for reminding the driver when the vehicle is out of the lane or, in case of the turning indicator light is switched off (Batavia, 1999). Emergency Braking System (EBS) is utilised to avoid the collision or mitigating the impact during critical situations by applying the braking system automatically, which has become mandatory for new heavy vehicles starting from 2013 (Okuda et al., 2014). As a conclusion, Table 1.1 summaries the different ADAS functions which have already been developed in market.

The automotive market is moving towards electrification of the car and autonomous driving to lower emissions, optimize traffic congestion and reduce other hazards. This trend needs ADAS electronic systems that can take decisions and acting in the place of

a human driver. These systems decide and act on safety applications such as steering, braking or transmission without causing injury to car passengers through wrong operations (Lopez & Clairet, 2016). With the advances in the safety requirements in automotive, advanced driver assistance systems, or ADAS, are becoming increasingly commonplace in all the modern automobiles. These systems provide the driver with the information about the surrounding and potentially automatically intervene with the process of driving to prevent accidents and road fatalities (Mody et al., 2017).

Table 1.1

ADAS Functions Overview.

ADAS Task	Description	Level	Impact
Adaptive Cruise Control	Automatic control speed and distance in relation to the heading vehicles	C	Lo
Traffic Sign Recognition	Recognizing the current road status to control the vehicle more suitable to the environment	I/S/W	Lo + La
Emergency Braking	Suddenly brake if any objects emerge into the minimum safety distance requirement	C	Lo
Pedestrian Detection	By utilizing radar or vision sensors to detect the locations of the pedestrians near the vehicle	W/S	Lo + La
Collision Avoidance	Warn driver in case of any collisions cause to the vehicle	C	Lo
Lane Departure Warning	Warn driver to stay in lane and providing information for potential collision when vehicle in adjacent lanes	W/S	La
Parking Assist	Assist driver parking the vehicle in the right place	I/W/S	La
Rear Collision Warning	Warn driver to avoid collision from backside	W/S	Lo
Surrounding View	Describe surrounding situation when vehicle in a non-dynamic environment	I/S	Lo + La
Intersection Negotiation	regulate vehicle traffic at intersections based on the Internet of vehicle controlled by computer at all scenarios	C	Lo + La
Fully Autonomous Driving	controlled by computer at all scenarios	C	Lo + La
Curve Assistance	Assist driver in keeping certain speed during curve maneuverer for safety issue	W/S	Lo
Navigation system	Provide vehicle position and calculate the optimal to the goal place	I/S/W	Lo + La
Vision Enhancement	Minimum and maximum operating temperature	I/W/S	Lo + La
Driver Monitoring	Limit for vibrations before damage or major fatigue occurs	I/W	Lo

Level: I=information, W=warn, C=control, S=support; Impact: Lo=longitudinal, La=lateral

For a typical autonomous or highly assisted driving scenario, dynamic obstacles are often combined with complex conditions such as driving in narrow streets, turning to avoid pedestrians, or braking for traffic lights. To achieve these tasks, several Heterogeneous sensors are used to monitor the surrounding environment deriving estimates of the physical state of targets. Multi-sensor fusion is used to reduce the uncertainty of state estimates and infer state properties which cannot be directly estimated. For instance, image sensors have drawbacks such as low ability to detect range information but are very efficient in appearance description. Radar provides precise range information but is limited in the representation of shape. Light Detection and Ranging (LiDAR) provides both appearance and range information but has issues with object segmentation. Due to the proprieties of the various sensors, a fusion based ADAS system becomes significantly important for the research activities. Figure 1.1 illustrates the ADAS functions with respect to the corresponding sensors in Texas Instruments.

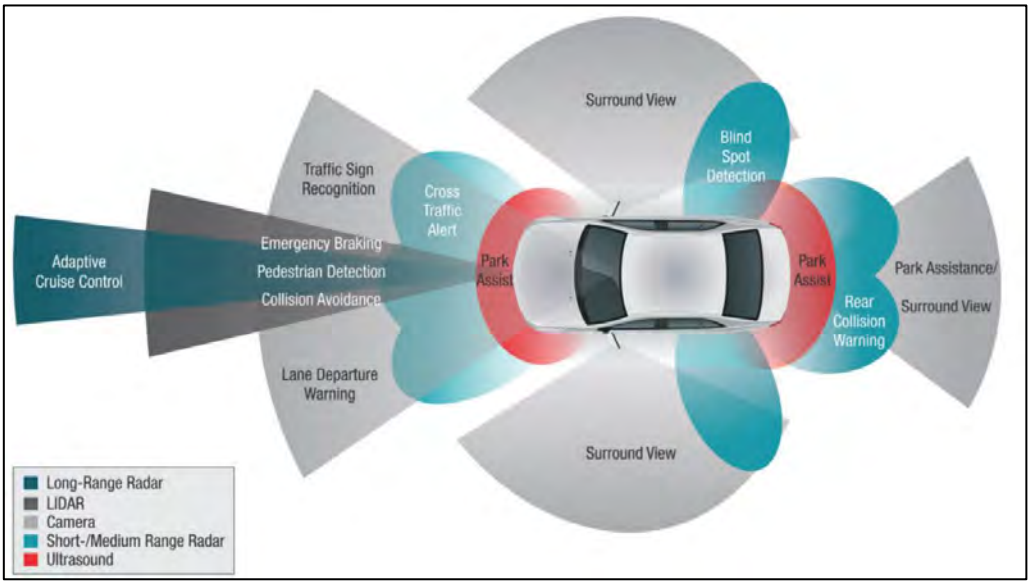

Figure 1.1. Data Fusion solution for ADAS by Texas Instruments (TI, 2013).

Most of ADAS works have been conducted within collaborative research programs involving car manufacturers, OEM, and research laboratories around the world. Recent research and development on highly autonomous driving aim to ultimately replace the driver's actions with robotic functions. The first successful steps were dedicated to embedded assistance systems such as Adaptive Cruise Control (ACC), obstacle collision avoidance or mitigation (Automatic Emergency Braking), vehicle stability control (ESC), lane keeping or lane departure avoidance (Gruyer et al., 2017).

1.2.2 Field Programmable Gate Array (FPGA)

FPGAs are reconfigurable computing devices contain many programmable units that can be used to solve specific computational problems (Giesemann et al., 2014). FPGA is a semiconductor-integrated circuit where a large majority of the electrical functionality inside the device can be changed, even after the equipment has been shipped to customers out in the field. These powerful devices can be customized to accelerate key workloads and enable design engineers to adapt to emerging standards or changing requirements (X. Wang et al., 2019). In contrast to application-specific integrated circuits (ASICs), which are only configured once during the manufacturing process, FPGAs can be reconfigured as often as needed. This allows improving and changing the architecture, applying bug-fixes, or using FPGAs to rapidly prototype hardware designs, which can later be manufactured as ASICs. FPGAs additionally allow reconfiguration on the fly to solve different tasks (Ziener, 2018). However, the reconfigurability and flexibility of FPGA come with the cost of design flow complexity,

and with less specialized components (i.e. floating point operations). Hence, the design must be implemented with general purpose logic by exploiting massive parallelism, typically in the form of deep pipelines to overcome this issue (Purkayastha et al., 2018).

ASIC style approaches can yield extremely high performance and great power efficiency but lack the highly desirable flexibility and software eco-system offered by Programmable Logic Devices (PLD) (Burgio et al., 2017). FPGAs have an important advantage over ASICs: they are reconfigurable, which gives them some of the flexibility of software (Velez et al., 2015). Furthermore, more complex ADAS applications are demanded; they do not need to do image processing only, but they also need to communicate with other devices and offer a usable user interface.

The hardware design flow of FPGA can be seen in Figure 1.2. The flow starts with hardware design specifications. In this stage, all the related technical specifications of the required design must be defined (Intel, 2020). These specifications include: the required hardware design functionality, speed, memory size, number of input/output peripherals, and data transfer protocols (Li et al., 2005). The next step would be the architecture level design, where the hardware design will be divided into system and sub-system modules. This will be achieved through the third step, which is the micro-architecture level design (Gerstlauer et al., 2009).

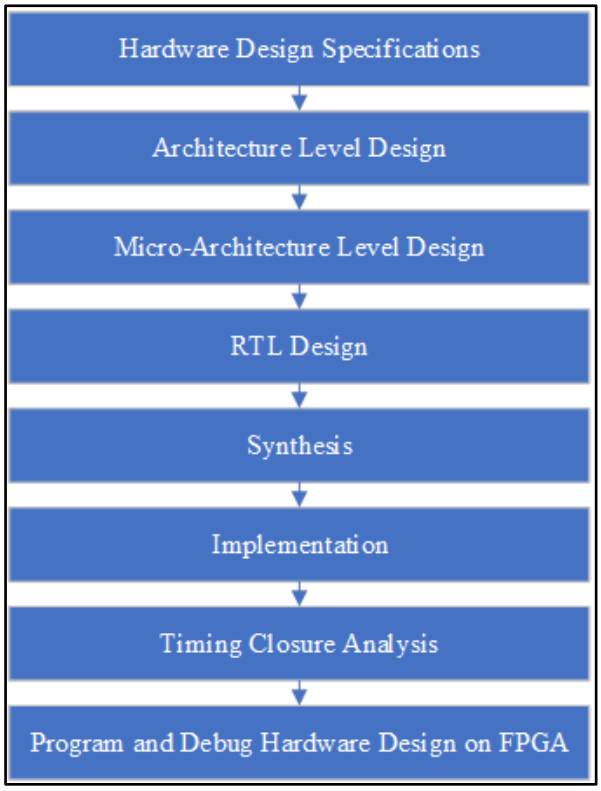

Figure 1.2. Hardware Design Flow of FPGA (Intel, 2020).

After finishing the architecture and micro-architecture level of the required hardware design, the Register Transfer Level design will start. In this level, the system and sub-system blocks will be translated to Hardware netlist through Hardware Description Language (HDL) (Intel, 2009). Once the digital design of the modules is done, synthesis and implementation processes start. These steps translate the HDL design into physical netlist ready for timing analysis. The timing closure analysis makes sure the hardware is functional from time perspective; the design meets the system speed requirements (Gerstlauer et al., 2009; Y. Wang & Nouta, 2004). Finally, the hardware design is ready to execute on FPGA.

1.2.3 Heterogeneous System Architecture (HSA)

Heterogeneous System Architecture (HSA) is a new computer platform infrastructure and associated software, that allows processors of different types and architectures to work efficiently and cooperatively in shared memory from a single source program (Kyriazis, 2012; Bauer et al., 2019). The utilisation of heterogeneous platforms to develop ADAS systems is important (Kocić et al., 2016; Kukkala et al., 2018), however, programming various devices in Heterogeneous platforms implementation requires high adaptivity to the currently available resources spawning work over heterogeneous CPU clusters is challenging. Furthermore, Programming Computational devices in Heterogeneous systems is challenging, due to management overhead, data transfer delays, and a missing unification of the programming flow (Kocić et al., 2016). In addition, the inherent architectural complexity of many-core platforms makes it almost impossible to derive real-time guarantees using "traditional" state-of-the-art techniques, ultimately preventing their adoption in real industrial settings (Burgio et al., 2017). Multiple platforms were found in literature review for ADAS designs based on Heterogeneous architectures, SoC-FPGA, CPU-GPU and GPU-FPGA platforms. This Heterogeneous platform improves the performance of embedded using hardware containing more than one type of processor. This approach has shown improved performance, particularly in the domain of artificial intelligence, in which computationally demanding models must be trained and executed. However, these computational environments pose various challenges to software engineering, since applications must be designed and developed differently while accounting for target hardware architectures that are inherently different (Andrade, Lwakatare, et al., 2019).

Guaranteeing real-time performance is crucial for state-of-the-art ADAS applications as it can provide as much time as possible for drivers to make better decisions in a relatively short time frame. This bound not only comes from the inherent time-criticality of ADAS tasks, but also stems from the demand to efficiently process the massive amount of data captured by the various types of sensors equipped in modern vehicles (X. Wang et al., 2019). As mentioned earlier, the development of several heterogeneous functions requires long time and high costs (Calefato et al., 2016). For example, Nvidia provides a solid solution for ADAS applications development, the AGX Pegasus development kit with 320 Tera Operations Per Second (TOPS) (Nvidia, 2020). This product provides high performance applications; however, the cost is high to afford for researchers. On the other hand, a low-cost solution like Arduino can be used by researchers to develop few ADAS applications. However, Arduino has a limitation in peripheral controllers; it has one controller for each interface, UART, I2C, and SPI (Louis, 2016). This is sufficient for a small-scale application, not for ADAS applications development which require heavy processing power (Sahlbach et al., 2010).

1.3 Problem Statement

ADAS has been investigated for many years among manufacturers, customers and the whole society due to its significance in enhancing driver performance, and increasing road safety (Hammond et al., 2015; Calefato et al., 2016; Mirnig et al., 2017; Yoshizawa & Iwasaki, 2018; Okamoto & Tsiotras, 2019). The development of ADAS

algorithms is a challenging task due to the high innovation rate and processing demands of applications (Giesemann et al., 2014; Kocić et al., 2016), and the inclusion of several heterogeneous functions and data sources provided by different manufacturers (Calefato et al., 2016).

The problem faced by today's automotive industry is that solutions implementing these ADAS-functions are becoming numerous, heterogeneous and provided by different suppliers, since different functions are using different sensors and vehicle actuators (Calefato et al., 2016). Furthermore, the design of an embedded system for ADAS applications is not straightforward, and several requirements must be addressed such as computational performance, cost, size, power consumption, platform flexibility, and time-to-market (Velez et al., 2015). ADAS development cycle starts with data acquisition, data preparation, test preparation, design and development, and system integration (Ball & Tang, 2019; Dell, 2019). Huge volumes of sensor data are captured by a fleet of test vehicles, which may comprise video sequences, ultrasonic, radar, LiDAR, Global Positioning System (GPS), and others Once the data has been ingested, the engineering teams will start to prepare the data which may include trimming, decoding, data enrichment (Dell, 2019).

The literature review showed a significant gap in determining and developing a data acquisition engine for ADAS applications based on sensor fusion approach. Most of the work presented in literature focused mainly on vision applications and algorithms. This is good for vision applications (Hammond et al., 2015; Hamza et al., 2012; Schumacher & Greiner, 2014; Velez et al., 2015; X. Wang et al., 2019), but not for

sensor applications. In addition, the proposed algorithms and systems in literature presented a specific ADAS functionality. There is a need to present system-level specifications for full ADAS system design. Apart from the mentioned limitations, these solutions are implemented based on single processing platform. It is not expected to develop a system with serial implementations to be speeded-up significantly on future processors as clock frequency is limited by power consumption and memory bandwidth (Ranft et al., 2011).

With rapidly evolving standards and requirements for ADAS applications, the need for flexibility and faster development cycles, while maintaining high performance is the primary concern for system designers (Bauer et al., 2019; X. Wang et al., 2019). FPGAs provide a promising approach to solve this problem since they can adapt their hardware based to the current needs of the ADAS applications with mid to high level of performance (Schwiegelshohn et al., 2015; Zhong et al., 2016). FPGA utilisation helps in gaining better performance compared with low-range cost systems. Furthermore, FPGA has plenty of variety the developer can select based on the available budget (Tan et al., 2019; X. Zhang et al., 2020). It can go as expensive as the high-end platforms, and as cheap as the low-end platforms. However, developing an application-specific FPGA is complex and requires more effort than implementing the same algorithm on the CPU or GPU (Besta et al., 2019), as seen in Figure 1.3. Heterogeneous platforms based on processor and FPGA have recently gained popularity for systems that demand the programmability and performance offered by a processor and configurability and flexibility achieved by the FPGA fabric (Zhong et al., 2016).

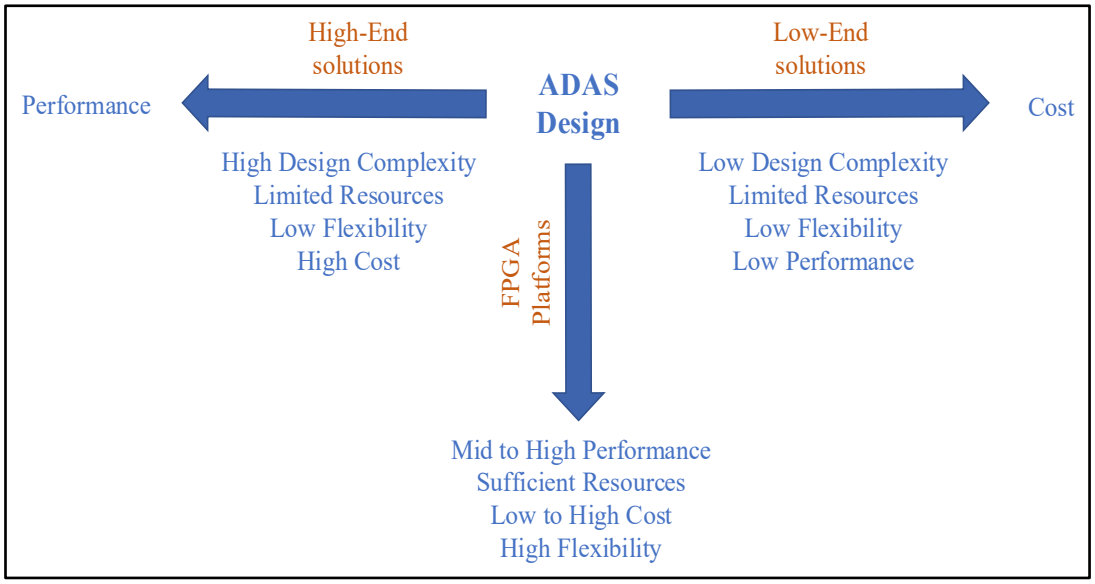

Figure 1.3. ADAS applications and Hardware Platforms.

1.4 Research Questions

1. What are the heterogeneous design techniques and trends available in the literature review?
2. Is it achievable to use a heterogeneous platform to implement ADAS data acquisition engine (ADAS-DAE)?
3. Is it possible to prototype ADAS-DAE on real-time basis?
4. What is the design flow required for SoC-FPGA ADAS-DAE design?
5. What are the differences between the proposed ADAS-DAE compared to the other designs in the literature review?

1.5 Research Objectives

The objectives of this research are:

1. To investigate and analyse the state-of-the-art ADAS engine design trends and categorize the related work to groups via systematic literature review.
2. To design a hardware-based ADAS Data Acquisition Engine (DAE) through the utilisation of SoC-FPGA heterogeneous computing platform.
3. To implement a real-time prototype for the proposed ADAS-DAE.
4. To derive a design flow guideline for the SoC-FPGA heterogeneous platform for ADAS-DAE.
5. To evaluate the proposed architecture of ADAS-DAE.

1.6 Research Scope

The scope of this research focuses on the hardware development of ADAS-DAE. This engine is used to acquire data from heterogenous sensors. The system design integrates FPGA with SoC to create a customizable heterogeneous platform that segments the system functionality into tasks. The validation of the proposed design is based on functionality tests, which has two levels of testing, indoor test for each component separately, and outdoor test when integrating the system on car.

The proposed design in this study utilised two development kits, DE10 Nano SoC FPGA development kit from Intel, and Raspberry Pi development board. The utilised boards will not impact the proposed system design since the design flow is the same for

any FPGA development board. Hence, section of FPGA is not important. In the system, there are two SoC sub-system, external one (Raspberry Pi), and internal one (ARM SoC). The external sub-system was used to provide the system with all the required augments for the ease of prototype implementation, like LCD, mouse, and keyboard. The internal SoC was part of the ADAS-DAE as the main system processor. Due to Movement Order Control (MCO) limitations and COVID-19 pandemic, the data collection was limited to the testing scenarios. However, the acquired data was enough to verify and validate the system functionality. The testing scenarios were based on four independent real-time tests to verify and validate the system functionality and reliability. The utilised sensors in this design are LiDAR, ultrasonic, and IMU. However, other sensors related to ADAS can be adapted to the system based on the functionality. These sensors were used as proof-of-concept to validate the system functionality. The proposed system can be integrated with any car, therefore, car selection in place is not important.

1.7 Significance of Study

Recently, ADAS development becomes very distinguished (Schwiegelshohn et al., 2015), and one of the most dynamically developing components in modern cars due to several reasons, which are road accidents reduction, and road safety improvement (Komorkiewicz et al., 2016). The need for more computational power leads to the usage of heterogeneous systems within today's embedded systems (Bauer et al., 2019).

This study aims to provide a heterogeneous SoC-FPGA ADAS-DAE for researchers to develop their ADAS algorithms and collect data with less complexity and standardized design flow. The proposed design allows both hardware designs and software developers to interact and develop their ADAS applications based on the proposed design methodology. The unified hardware and software flows reduces the heterogenous design cycle complexity. Furthermore, it provides high level of flexibility in terms of hardware resources requirements. The cost of the proposed design can vary based on the available budge and required functionality. hence, there is a flexibility in selecting the appropriate development kits and sensors. The unified development flow provides several options in terms of programming languages. From the hardware perspective, the designer can select the preferred HDL, either Verilog, systemVerilog, or VHDL. Furthermore, the embedded Linux availability in the system provides wide range of programming languages for software development.

The hardware limitations in the proposed design were minimised through the utilisation of FPGA. The designer can add the preferred sensor with any data transfer protocol, like Inter-Integrated Circuit (I2C), Universal Asynchronous Receiver-Transmitter (UART), General Purpose In/Out (GPIO), and Serial Peripheral Interface (SPI); all the data transfer protocols can be easily implemented on FPGA. Furthermore, the proposed design methodology reduces the complexity in SoC-FPGA heterogenous platform by adding middleware layer for software developers to interact with the FPGA system in the form of application program interface.

1.8 Operational Definitions

1. **FPGA:** it is a reconfigurable computing device contain many programmable units that can be used to solve specific computational problems (Besta et al., 2019).
2. **SoC:** the embedded System-on-Chip is generally defined as special purpose computer-systems designed to perform one or more dedicated functions, usually with real-time constraints, and have computer hardware and software embedded as parts of a complete device or system (Abid et al., 2018).
3. **HSA:** Heterogeneous System Architecture is a computing platform that contains more than one type of processing unit. Heterogeneous platforms may contain, for example, a combination of multi–core Central Processing Unit (CPU), Graphics Processing Unit (GPU) and FPGA, which were initially proposed to execute tasks of specific types (Andrade, Schroeder, et al., 2019).
4. **ADAS:** it is a collection of numerous intelligent units integrated in the vehicle itself. All these units perform different tasks and assist the human driver in driving (Rahul, 2016).
5. **System Architecture**: fundamental concepts or properties of a system in its environment embodied in its elements, relationships, and in the principles of its design and evolution (International Organization for Standardization, 2011).
6. **Quartus Prime**: it is a software compilation tool from intel used to achieve the hardware design of intel FPGA. This tool provides the complete flow from design, synthesis, timing analysis, and FPGA configuration (Intel, 2017).

7. **Embedded Linux**: for embedded systems, Linux operating system can be utilised to run real-time application. It uses software from many other projects in order to provide a complete operating system that fully utilise the system functionality (Sally, 2010, pp. 1–23).
8. **LiDAR:** Light Detection and Ranging is a standard tool for collecting accurate and dense topographic data at extremely high speed. These data have found use in many applications and several new applications are being discovered regularly (Lohani & Ghosh, 2017).
9. **Ultrasonic:** it is a sensor uses sound waves above 20 khz range to detect objects in proximity. Ultrasonic sensor is prevalent for ADAS applications, especially short range applications like parking assistance, 4–16 sensors are used to detect obstacles when parking a vehicle (TI, 2019).
10. **IMU:** it is an electronic device that uses a specific body, angular rate and magnetic fields surround the body to calculate the orientation of an object in space. It uses a combination of accelerometer sensor and gyroscopes sensor to calculate the roll, pitch and yaw of an object (Faisal et al., 2020).
11. **Yocto:** The Yocto Project is an open-source collaboration project focused on embedded Linux development. The project currently provides a build system that is referred to as the OpenEmbedded build system. The Yocto Project provides various ancillary tools for the embedded developers to build their embedded Linux systems with various features and functions (Rifenbark, 2014).

12. **Nios II:** the Nios II processor is a general-purpose processor with full 32-bit instruction set, data path, address space, 32 general-purpose registers and 32 interrupt sources (Intel Corporation, 2015).
13. **Process Core:** it is the proposed hardware design element in this research. It is a combination of Nios II processor, on chip memory, I2C controller, UART controller, and general-purpose input/out.
14. **ADAS-DAE:** this is the proposed hardware design of ADAS Data Acquisition Engine based on heterogenous SoC-FPGA platform. This design was validated and tested in this study to provide it as a reference solution of ADAS applications.
15. **Design Methodology:** all the design steps, flows, actions, and decisions which are considered during the design cycle of the proposed design and implementation.
16. **DE10-nano:** this is the SoC-FPGA development board provided by Intel. This board is the base design environment to design and implement the proposed design of this study.

1.9 Thesis Layout

This search composed of six chapters, and Table 1.2 present the main goal of each chapter. A brief introduction and overview of the research are provided in Chapter one. The problem statement, research objective, questions, scope, and significance of the work, are present.

Chapter Two: In Chapter Two, an in-depth investigation was conducted about the previous studies in ADAS hardware development field. A systematic review protocol is developed to collect the literature and to analyse the challenges and produce a taxonomy for the research resources.

Chapter Three: In this chapter, the proposed system and design steps that needed to develop an ADAS hardware engine based on Heterogeneous platform are presented.

Chapter Four*:* this chapter presents the technical aspects and tutorials needed for the design flow of the proposed ADAS engine. The design was achieved through hardware and software flows, which are both reported in this chapter.

Chapter Five: the results of the proposed design tests are reported in this chapter. Furthermore, the data collected from sensors was analysed in this chapter with discussions.

Chapter Six: in this chapter, the summary of the research findings, contribution, claims, and comparative analysis were reported.

Table 1.2

List of the goals of each chapter in thesis.

Chapter	Research Goals	Related Objective
One	Introduces the research topic, brief background, of the problem statement, and research objectives and define the scope of this research.	Objective I
Two	Surveys the literature in ADAS hardware design area, taxonomized the information, identifying gaps in current knowledge, showing the weaknesses, challenges and points of view and critically analyse the gathered information.	Objective I
Three	Presents the proposed design techniques and procedures, sensors connections, testing scenarios, and experimental setup required to run and test the proposed ADAS engine.	Objective II Objective III
Four	Presents all the technical aspects of the design flow of ADAS engine. Furthermore, all the steps needed to compile the design from hardware flow to software flow are presented in tutorial format.	Objective IV
Five	Presents the test scenarios output, the data acquired by the system and its analysis.	Objective II Objective III
Six	Summary of the main findings, research contributions, study limitations, and future work, of this research.	Objective V

1.10 Summary

In this chapter, a detailed overview about the research topic was presented in three parts, ADAS, FPGA, and heterogenous system design. The problem statement of this study was illustrated. The research objectives and research questions were presented in this chapter. Furthermore, the scope and significance of this study were explained. Few terms were defined in the operational definition section. Finally, the thesis layout and the relation between each chapter and research objectives were discussed.