ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization) Vol.2, Special Issue 2, May 2014

International Conference On Advances in Computer & Communication Engineering (ACCE - 2014)
on 21st & 22nd April 2014, Organized by
Department of CSE & ISE, Vemana Institute of Technology, Bengaluru, India

Performance Efficient Soft Processor for Embedded Graphics

Adesh Panwar

Bangalore, India

ABSTRACT: Field programmable gate arrays (FPGAs) are quick solutions to create hardware circuits. With the increasing capacities and decreasing costs of these devices has enabled designers to readily incorporate them in their designs. This paper presents a study on designing a graphics generator using FPGA.

KEYWORDS: Embedded Graphics, Soft Processor

I. INTRODUCTION

Graphics is derived from the word graph which is a visual presentation using x and y axis on some surface to inform or illustrate. Such information is created in digital word and is seen on a digital screen. The output on any such digital device is termed as graphics. Embedded system is built for specific requirement and performs pre-defined tasks, such systems are provided with limited number of user interface and one such interface is through visual representation using graphics. Due to its availability in various form factors and low cost the LCD's has become imperative for interaction between the operator / user and the product. Such systems are essentially provided with some embedded graphics processor or core to generate and control graphics. One such use of embedded graphics generator is found in the implementation of Global Positioning System (GPS) for applications like Autonomous Navigation Systems, Fleet tracking, Map matching etc.

This paper proposes a soft processor for generation of embedded graphics for navigational aid. Section II introduces the basic block diagram of such system and its interface with GPS. A literature survey on the important criterion of design is presented in Section III. Section IV presents some conclusion.

II. PROPOSED SYSTEM

Figure1, shows the architecture of a typicalNavigation system, wherein a dedicated Processor acquires current location through a GPS receiver[1],interpolates it on a Map and this information is displayed to the user. Design of such a system will require Navigation Data input source, a platform for the graphics generation and Soft Processor core to execute the front end application.

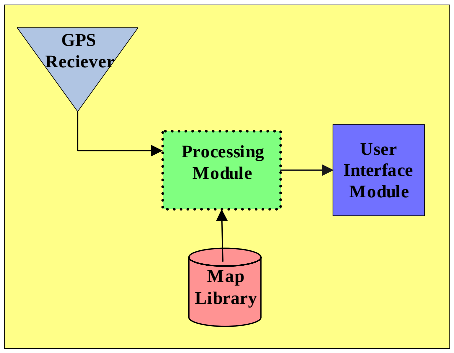

Figure 1: Typical Navigation System

Therefore, the discussions in this paper are limited to following three core areas:

- *Navigation*
- *Graphics Generation Platform and*
- *Soft processor core*

III. LITERATURE SURVEY

A. Navigation

The most basic function of a land vehicle navigation system is to accurately identify the location of a vehicle with respect to a given coordinate system. Such system works using an on-board computer that continuously collects data from multiple sensors mounted inside the vehicle [1]. The computer uses this data to compute the vehicle's location by interpolating, applying various fixes and representing the location to the driver by means of an electronic interface.

A simple functional schematic diagram representing the operation of a land vehicle navigation system is shown in Figure 1. Purpose of such a system is to locate the user on 3D space, but a such a system using only GPS cannot continuously position the vehicle [2], so other navigation aids are necessary [3] [4] [5].

Tall buildings, dense foliage, or uneven terrain, stands between a GPS receiver and a GPS satellite to block the signals. Thus, in heavily foliated environments a GPS receiver may not be able to provide a position fix for indefinitely long periods of time. Even if GPS position fixes are available they may contain errors and are not very accurate [6]. Due to selective availability (SA) [1] the signals from GPS satellite can be intentionally degraded or only limited users may be allowed to access the signals without SA. Solution to such situation is to include some combination of sensors.

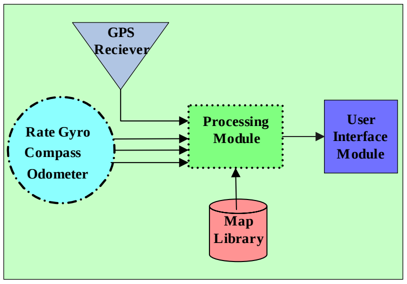

Figure 2: Block Diagram with Additional Sensors Input

Figure 2 shows the Navigational System with additional sensors input. Low-cost gyroscopes, compasses, an odometer, inclinometers, and/or accelerometers which can accurately measure changes in the vehicle's position over a short time periods can used along with a GPS receiver and can support navigation when GPS signals are unavailable.

B. Graphics Generation Platform

The time critical application platform faces challenges to maintain cost effective solutions to new visually appealing functionality such as data fusion, synthetic vision, and maps [7]. COTS graphics processing units (GPUs) are one solution to the above problem. GPUs provide the high-performance requirements, but these devices were not originally designed for time critical application wherein the device is exposed to high temperature, power constraints, long lifecycle support, and various certification requirements.

Because of the complexity and criticality of GPUs in time critical applications, it is believed that COTS GPUs require special attention [8]. The original markets for COTS GPU development have typically been portable and automotive industries. There has been plenty new architectures release at a remarkable rate to keep pace with the competition. But, all these designs are not readily supported in following years.

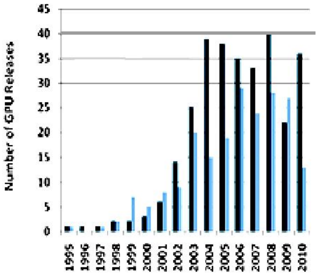

Figure 3: Number of Released GPUs

Figure 3 [2], shows that in the past sixteen years, over 500 GPUs have been released by such industries [9] [10]. Because of this nearly continual product release cycle, each device is only available for a very limited time, which certainly does not support the years or decades of lifecycle support required in time critical applications. Some other drawbacks with COTs based GPUs are:

- *Functionalities offered are beyond the scope of final product.*
- *System on Chip solutions is not cost effective with low volumes.*
- *High performance requirement of time critical applications are not meet.*

The actual graphical requirement of a time critical application lies between high-end GPUs and the low-end SOC GPU cores. If the graphical processing are designed using a FPGA then it will provide higher performance, longer life cycle and lower cost.Advantages of using FPGA for graphic application are:

- *Functionality*
- *Environmental Considerations*
- *Life Cycle Constraints*

1) Functionality: The pure amount of information available to the operator today requires careful attention to the human factor issues in order to maximize situational awareness. With the increasing application complexity and availability of multiple sensors has compelled designers to use custom solutions. Few examples are:

- *Display with Panoramic View*
- *Synthetic Vision on Primary Flight Display*
- *Engine Gauges on Multifunctional Display*

Designing custom functionalities with faster computations is a core advantage with FPGA.

*2) Environmental Considerations:*Subjecting commercial-grade parts in industrial-grade applications can result in lower reliability of the component and decrease in performance & throughput. With the improvement in technology many programmable chips such as a FPGA is now available in industrial and space grade operational environment. This offers a considerable advantage in terms of reliability and performance compared to a commercial grade COTs GPU.

*3) Life Cycle Constraints:*Time critical application equipment is expected to be in service for few years whereas consumer goods are in market for only few months. Therefore, when time critical application equipment is based on consumer technology devices, the life cycle support becomes challenging.Secondly, the design and certification cycle time for such applications is very high, so it is meaningful to design products that can survive many years of production. The longer that the individual components are available, the less reinvestment required for that product.Design using FPGAs provide advantage of code reusability over variety of device families and vendors. Products with obsolete devices can be supported with re-targeted code on new generation devices.

C. Soft Processor Core

FPGAs with processor provides software implementation platform along with the custom logins. Such devices are advantageous in terms of:

- *Reduced parts costs*
- *Smaller board sizes, and*
- *Improved system performance due to shorter logic path between processor and FPGA*

There are two types of processors available on a FPGA chip: Hardcore and Soft Processor Core. The major difference between these two is that a hard core processor has to be fabricated onto the chip with other configurable logic fabric [11] whereas a soft-core processor is synthesized onto the FPGA's fabric with other custom logics.

Soft Processor Cores have advantage of utilizing standard mass manufacturing techniques and hence lower cost. They offer higher flexibility to designers for implementing custom number of cores depending upon FPGA size. However, Soft Processor Core has some limitations [12]:

- *Reduced processor performance,*
- *Higher power consumption,*
- *Larger size*

FPGA vendors have introduced Soft Processor Cores specifically targeted for FPGA implementation. These processors have features specifically tailored for efficient use FPGA resources such as:

- *Instruction Sets,*
- *Arithmetic Logic Units (ALU),*
- *Register*

With this the performance overhead of Soft Processors Core compared to Hard Core Processors (such as Application-Specific Integrated Circuits) has significantly improved [12].

FPGA Soft Processor Core comes with application development tools know as Integrated Development Environment (IDE). It enables features section, cache instantiation, setting of parameters etc. A different problem of developing custom data path units and accompanying custom instructions is faced by the parameterized soft cores [12]. Figure 4 shows the soft processor core development platform.

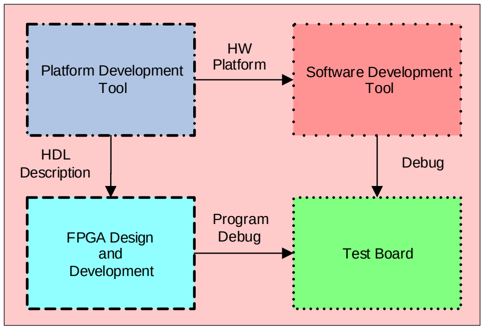
Platform Development Tool
HW Platform
Software Development Tool
HDL Description
Debug
FPGA Design and Development
Program Debug
Test Board

Figure 4: Soft Processor Core Development Platform

There are many types of Soft Processor Core available in market, few of them are:

- *Altera Nios/ NiosII,*
- *LatticeMico32*
- *Xilinx Micro Blaze*

They offer memory and logic elements with several Intellectual Property (IP) peripherals for the rapid development of System on Programmable-Chip (SoPC) [13].

The basic selection criterions for Soft Processor Core are:

1) Power and Performance: Two important factors to be considered during system design are Operational Performance and Power requirement. Product designers have to make a tradeoff between these two parameters for implementing the desired system functionality.

2) Integrated Development Environment Tools: Effective tool evaluation and analysis is important, following factors can have significant effect on design cycle efficiency [15]:

- *Ease of use and feature set*
- *Design tool flow*
- *Development environment tool maturity*
- *Compatibility between major software releases*
- *Available training and quality of tool tutorials*
- *Debug and verification capabilities*

3) Operating System Considerations: Another important design factor is the ability to utilize popular operating systems (OSs). Most embedded designs on 32-bit processors include an OS to reduce the design time of the software by providing an abstraction interface level to the software. Most operating systems include the OS and any lower-level software required to connect the OS to the hardware. Board Support Package (BSP) refers to these collections of software elements. The BSP can include items such as the processor boot code and interrupt service routines for peripherals.

IV. CONCLUSION

Literature survey on Performance efficient soft processor for 2D graphics generator is conducted and following conclusion is derived:

A. Navigation

Use of Dead Reckoning systems (DRS) along with a GPS receiver provides an improved performance in the application. DRS are capable of generation information in absence of GPS signals and can be used for Navigational aid.

B. Graphics Generation Platform

A FPGA based GPU provides tremendous lifecycle, certification, and flexibility advantages for critical applications. Because FPGAs are not closely tied to consumer applications, they are usually available for longer duration as compared to discrete GPUs. Furthermore, if the FPGA device does go obsolete, the GPU design could be retargeted on an alternate FPGA with simple modifications.

C. Soft Processor Core

A popular soft core processor from Altera's is NIOS II. It offers following features:

- *Load- Store RISC architecture*
- *Customized architectural parameters*
- *Data bus width selection between 16 or 32 bits*
- *Selectable register file sizes and cache size*
- *Custom instructions for user defined*

It has NIOS II Integrated Development Environment (IDE) to build, run, and debug software of several platforms. It also provides a SOPC builder [14], which assists in rapid development and evaluation (verification / validation) of embedded systems. It offers capabilities to integrate intellectual property (IP), custom components and ready to use peripherals on a single platform to enabling reduced design time [13].

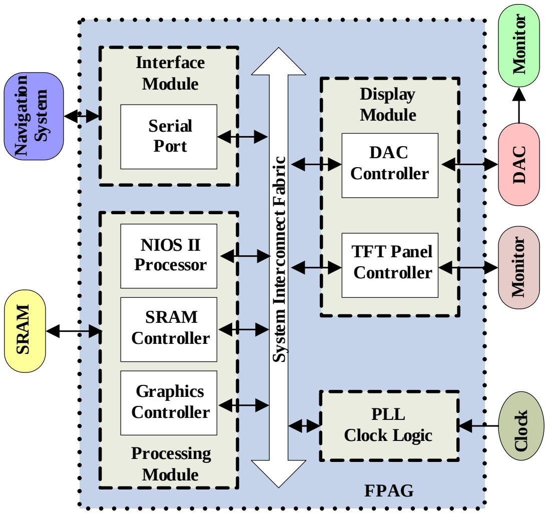

Figure 5: Detailed Block Diagram

A detailed block diagram is shown in Figure 5 wherein the multisensory Navigational input data is received by FPGA based Graphics Processing Unit and the custom application is supported on a soft processor core. Various other modules such as Memory interface and display interface are the part of FPGA.

REFERENCES

1. Alberto Trombetta and Wei Jiang (2011), ‘Privacy-Preserving Updates to Anonymous and Confidential Databases’, IEEE Transactions on Knowledge and Data Engineering, Vol. 22, pp. 578-568.
2. Ahmed El Rabbany, “Introduction to GPS”.
3. U. Iqbal, J. Georgy, M.J. Korenberg, A. Noureldin, “Modeling Residual Errors of GPS”.
4. D. A. Divis, “GLONASS emerges; Change in ISNS game plan,” GPS World, vol. 7, no. 5, p. 12, May 1996.
5. B. W. Parkinson, J. Spilker, Jr., P. Enge, and P. Axelrad,Eds., Global Positioning System: Theory and Applications, vol.2. Washington, DC: American Institute of Aeronautics and Astronautics, 1996.
6. R. L. French, “The evolution of automobile navigation systemsin Japan,” in Proc. 49th Annu. Meeting Institute of Navigation,June 1993, pp. 69–74.
7. Eric Abbott and David Powell, “Land-Vehicle Navigation Using GPS”.
8. Marcus Dutton, David Keezer, “The Challenges of Graphics Processing in the Avionics Industry”.
9. Certification Authorities Software Team, “Use of COTS Graphical Processors (CGP) in Airborne Display Systems”, Position Paper 29, Feb 2007.
10. “Comparison of ATI graphics processing units,”[Online]. Available: http://en.wikipedia.org/wiki/List_of_ATI_cards.
11. “Comparison of NVIDIA graphics processing units,” [Online]. Available: http://en.wikipedia.org/wiki/List_of_NVIDIA_Graphics_Processing_Units.
12. PetarBorisovMinev, ValentinaStoianovaKukenska, “Implementation of Soft Core Processor in FPGAs”.
13. Sheldon, D., R. Kumar, F. Vahid, R. Lysecky, D.Tullsen, “Application-Specific Customization Of Parameterized FPGA Soft-Core Processors”, International Conference on Computer-Aided Design, ICCAD, San Jose, November 2006.
14. Calderón, H., C. Elena, S. Vassiliadis, “Soft Core Processors and Embedded Processing”, Proceedings of Pro RISC, Veldhoven, The Netherlands, November 2005.
15. “SOPC builder Handbook”, http://www.Altera.com.
16. Cofer, R.C., B. Harding, “FPGA Soft Processor Design Considerations, Programmable Logic Design Line”, October 12, 2005.
17. “Cyclone III Device Handbook”, http://www.Altera.com.
18. “Quartus II Version 7.1 Handbook”, http://www.Altera.com.
19. “NIOS II Processor reference handbook”, http://www.Altera.com.