AD

AD-E404 275

Technical Report ARWSE-TR-19002

MULTICORE CONSIDERATIONS FOR SAFETY-CRITICAL SOFTWARE APPLICATIONS

36TH INTERNATIONAL SYSTEM SAFETY CONFERENCE

Brian Connell

February 2021

U.S. ARMY COMBAT CAPABILITIES DEVELOPMENT COMMAND ARMAMENTS CENTER

Weapons and Software Engineering Center

Picatinny Arsenal, New Jersey

The views, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

The citation in this report of the names of commercial firms or commercially available products or services does not constitute official endorsement by or approval of the U.S. Government.

Destroy by any means possible to prevent disclosure of contents or reconstruction of the document. Do not return to the originator.

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-01-0188*

1. REPORT DATE (DD-MM-YYYY)	2. REPORT TYPE	3. DATES COVERED (*From - To*)
February 2021	Final	August 13, 2018 to August 17, 2018

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
Multicore Considerations for Safety-critical Software Applications 36th International System Safety Conference	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHORS	5d. PROJECT NUMBER
Brian Connell	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
U.S. Army DEVCOM AC, WSEC Armament Software Engineering Center (FCDD-ACW-SV) Picatinny Arsenal, NJ 07806-5000	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR'S ACRONYM(S)
U.S. Army DEVCOM AC, ESIC Knowledge & Process Management Office (FCDD-ACE-K) Picatinny Arsenal, NJ 07806-5000	11. SPONSOR/MONITOR'S REPORT NUMBER(S) Technical Report ARWSE-TR-19002

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

The popularity and ubiquity of multicore processors and system on a chip (SoC) architectures present developers of safety-critical applications with special considerations in attempting to ensure against data corruption and assure predictable, error-free execution of safety-critical tasks. The issues may be further complicated if mixed-criticality tasks are housed on the same processing platform. In order to realize the energy, space, and performance dividends of multicore technology, developers of embedded safety-critical applications must address the limitations of multicore SoCs that threaten the certification, determinism and integrity of mixed-criticality applications. This report explores the state of the art use of multicore chipsets and SoC architectures, identifies the limitations of this technology for safety-critical applications, and describes a number of proposed solutions. It includes a summary of personal recommendations and experiences that may help computer scientists and software engineers leverage multicore processors without compromising system safety.

15. SUBJECT TERMS

Multicore Safety-critical applications Software safety

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Brian Connell
U	U	U	SAR	11	19b. TELEPHONE NUMBER (Include area code) (973) 724-1922

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

CONTENTS

INTRODUCTION

The popularity of the multicore processor is drawing added attention to the challenges posed by the application of this vital technology to safety-critical software applications. The advantages of multicore processors, in the context of size, weight, and power (SWAP) consumption, are increasingly clear. However, problems with certification of safety-significant applications justify re-examination and development of practical solutions before the industry is ready for widespread adoption in avionics, automotive, medical, and industrial control systems. Among the challenges posed by multicore processors are temporal determinism and control of error propagation. Present-day multicore devices are designed for nominal operating conditions and are not adequately partitioned to ensure against error propagation between safety-significant and non-safety functions. In order to gain a foothold in safety-related and safety-critical applications, multicore chip architectures must address the limitations in design abstraction and establish well-defined messaging interfaces with synchronized, predicable timing.

This report is intended to raise awareness about the obstacles facing broader adoption of safety-significant applications and the diverse set of tools and techniques being applied to these problems in academia, industry, and other partnerships. The author is seeking to provide expertise in safety certification of military applications and stimulate thought and dialogue with the hope that newly-developed solutions and architectures reflect a holistic approach well-suited for diverse applications in the defense and private industries.

The report begins with a brief discussion of the advantages of multicore technology and it also summarizes the limitations of present-day multiprocessor system on chip (MPSoC). An in-depth look at the broad array of solutions under evaluation is also presented. The author shares personal insights and updates about the use of multicore processors in military close-combat and active protection solutions. The findings of contemporary prototype and demonstrator systems are discussed. Then, conclusions are drawn regarding the efficacy and shortcomings of proposed solutions. Finally, suggestions are offered for experimental designs that address gaps in the amenability of this technology to safety-significant applications and certifications.

MULTICORE STRENGTHS AND LIMITATIONS

Migration to multicore platforms is driven by continued demand for increased processing speed. In a workshop called “Validation and Verification of Safety-Critical Software Applications on Multicore Processing Platforms” in Dahlgren, VA, on 12 June 2018, a representative from the Software Engineering Institute at Carnegie Mellon University spoke of the “power wall” that led to the cancellation of the Intel “Tejas” central processing unit (CPU) program. This was caused by overheating issues and prompted the research and development of dual-core technology (ref. 1). In the same discussion, a “memory wall,” characterized by a 2007/2008 Sandia National Laboratories (Albuquerque, NM) study on memory bandwidth, was cited as an additional limitation of conventional computing devices.

Single core processors incur diminishing returns with increased size. Multicore processors on the other hand, realize a linear increase in speed with size that are limited only by Amdahl's Law which defines size restrictions based on the non-parallelizable fraction of the application hosted on the multicore chip (ref. 2). Fortunately, many of the applications that employ multicore technology are amenable to parallelization and are comprised of unique functions that exchange data generally non-sequential in nature. Avionics and automotive subsystems are examples of such systems. For this reason, heterogeneous functions play well with MPSoC platforms. Efficiencies are achieved by allocating unique functions like signal processing, encoding, and cipher implementation onto dedicated cores with shared resources. Aggregation of disparate functions onto a single computing device reduces cost, energy, and envelope size. Cost savings extends to manufacturing and operating expenses. The reduction in overall components also promotes increased reliability.

The enabling design and architecture of multicore chipsets presents a number of challenges for developers and certifiers of safety-related and safety-critical components. Safety-critical tasks, in particular, often require temporal determinism with timely responses under worst-case conditions. The spate of automotive air bag accidents and fatalities in the past two decades underscores the importance of temporal compliance. Existing multicore architectures are designed and optimized for nominal operating conditions. The abstraction of memory architecture and addressing makes the prediction of real-time behavior difficult.

The propagation of errors between safety-critical and non-safety tasks is another concern facing today's multicore architectures. Without an architectural solution, developers are forced to partition these components and the memory they share and custom-certify them according to their individual MIL-STD-882E software criticality indices (SwCI). Alternatively, the entire application is certified to the standards of the most critical component. The latter option is often infeasible or cost-prohibitive. Given the present limitations of MPSoCs, certifying authorities view multiple processors, shared data and interaction between various communications busses as devices with increased complexity, making certification an expensive proposition.

PROPOSED SOLUTIONS FOR MULTICORE AND SAFETY-CRITICAL APPLICATIONS

The design goals established by the ACROSS MPSoC (ref. 2) academic/industry partnership in Europe give developers of multicore platforms cause for hope. However, the implementation and success of related demonstrator systems will be the ultimate predictor for expansion of the adoption of the solutions under consideration. Certifiable goals seek to ensure that failures of the most critical components (or SwCI) occur less than once in 100,000 years. Temporal determinism objectives include guaranteed execution deadlines for safety-critical tasks. Proposed designs will ensure that mixed-criticality applications are able to reside on a single MPSoC without interference or error propagation. Ultimately, an ideal design would support independent and "permanent" certification, obviating regression on previously certified and unchanged components.

The architectural and infrastructure solutions for the design goals summarized in this report are grounded in increased design abstraction, well-defined messaging interfaces and the replacement of interconnected cores with micro-components. The micro-components required would behave like a node in a distributed system and rely on time-triggering for temporal determinism. The proposed approach is known as a time-triggered network-on-chip (TTNoC). Such a network would be supported by a Trusted Research Manager (TRM) to adapt changes in communications bandwidth and component interactions (ref. 2). Another solution to the determinism challenge, proposed by Mr. King (ref. 3), is the technique of “slack scheduling,” which relies on threads that execute quicker than projected worst-case times. This allows reclamation of unused processor bandwidth by the real-time operating system (RTOS) and adaptive redistribution to other threads. King also suggests cache partitioning or allocation of physical memory to each core, as a means of reducing memory access bottlenecks.

The ARTEMIS ACROSS project in Europe, which has spawned ACROSS MPSoC prototypes and demonstrator platforms for automotive, aerospace, and industrial controls (ref. 2), is among the most comprehensive efforts to address the shortcomings of multicore chipset for safety-critical applications. However, there are a number of targeted approaches that have been found to be very serviceable for the given applications. Personal experience with networked munition systems for close-combat applications has shown that effecting a shutdown of secondary cores while running safety-critical tasks on a single core may be both practical and effective (depending on the relative proportion of safety-critical components). For the application in question, processor cache flushing is another consideration in the assurance of the integrity of any data that may be shared by safety-critical and non-safety-critical tasks. Active protection systems, which employ high-speed communications and servo loops for sensing and providing a directed response to incoming threats, may employ a hypervisor. This may be part of a layered partitioning approach that also uses the operating system for process isolation and the application for module isolation.

Impediments to certification of safety-critical applications on multicore technology may be tied to memory and run-time overhead associated with the instrumentation of code. A number of certification tools such as TBvision, use static code analysis to determine the best locations for probe points (ref. 4). The reduction in probe points provides a reduced memory footprint and certification approach amenable to test automation and hardware stubbing. In this approach, calls to operating system and library functions that manage memory are eliminated to prevent run-time concurrency issues.

The global adoption of multicore technology for embedded applications presents an additional challenge for integrators and software developers seeking to port their applications from legacy single core platforms to multicore platforms. In most cases, considerable expertise is required to migrate previously validated software to a new operating system and computing device. In 2015, Macher, G. and other authors advocated in their article the use of a migration pattern catalog (ref. 5). The approach described in this publication (which is focused on embedded automotive applications) enables proof of temporal determinism and ensures compliance with worst-case timing requirements. It also provides for more optimal use of multicore–specific features. The migration patterns addressed in the publication range from device and operating system selection to memory organization. Among the issues that were analyzed are the adaptation of software architectures designed for single core processors to leverage multicore parallel execution features. Recommendations included the use of dedicated synchronization primitives to guard against concurrent access to shared resources. Inter-core communication and synchronization patterns address the time penalties associated with synchronization of cores.

CONCLUSIONS

When suitably designed and architected, safety-critical software applications deployed with multicore technology may realize linear increases in speed, as gains are no longer possible with the simple addition of processors. The most successful solutions include a comprehensive approach to the challenges of temporal determinism, error propagation, and certification. The industries and applications that best benefit from multicore technology are comprised of heterogeneous functions that may share resources and are not time-sequenced generally.

The recommended approach for temporal determinism is the time-triggered network-on-chip (TTNoC), which relies in turn on a Trusted Research Manager (TRM). This approach requires an architecture where micro-components mimic the behavior of nodes in a distributed system and enables adaptations to the changes in communications bandwidth and component interactions. Slack scheduling (bandwidth recovery) and cache portioning (memory allocation) are additional techniques employed for improved determinism. In some instances, experience with safety-critical defense applications indicate that a shutdown of secondary cores during execution of safety-critical tasks is an effective way to ensure against error propagation through data stores shared by safety and non-safety significant functions. Cache flushing may be employed for added assurance.

Alternatively, a hypervisor may be used as part of a virtualization approach. This soft-partitioning approach allows the application to use memory, processing, and operating system resources that behave like a dedicated computing device. Challenges in certification of multicore-based safety-significant applications are being met in several ways, ranging from nonintrusive tools that minimize the time and memory overhead penalties to those that use migration pattern catalogs for tracking and monitoring critical choices such as chip and operating system selection during application porting.

REFERENCES

1. De Niz, D., “Multicore Processor Issues for Real-Time Systems,” Software Engineering Institute, January 2018.

2. El Salloum, C., Elshuber, M., Höftberger, O., and Isakovic, H., “The ACROSS MPSoC - A New Generation of Multicore Processors Designed for Safety-Critical Embedded Systems,” IEEE, Izmir, Turkey, September 2012.

3. King, T., “Solving the Processor Challenges for Safety-Critical Software,” Military Embedded Systems, October 2011.

4. Thomas, J., “New Technology Helps Multicore Meet Safety-Critical Standards,” Military Embedded Systems, May 2016.

5. Macher, G., Höller, A., Armengaud, E., and Kreiner, C., “Pattern Catalog for Multicore Migration of Embedded Automotive Systems,” Association of Computing Machinery, July 2015.

DISTRIBUTION LIST

U.S. Army DEVCOM AC
ATTN: FCDD-ACE-K
FCDD-ACW-SV, B. Connell
A. Hatley
Picatinny Arsenal, NJ 07806-5000

Defense Technical Information Center (DTIC)
ATTN: Accessions Division
8725 John J. Kingman Road, Ste 0944
Fort Belvoir, VA 22060-6218

GIDEP Operations Center
P.O. Box 8000
Corona, CA 91718-8000
gidep@gidep.org

REVIEW AND APPROVAL OF ARDEC REPORTS

THIS IS A:

☒ **TECHNICAL REPORT** ☐ **SPECIAL REPORT**

☐ **MEMORANDUM REPORT** ☐ **ARMAMENT GRADUATE SCHOOL REPORT**

FUNDING SOURCE ARDEC Armament SEC Overhead
[e.g., TEX3; 6.1 (ILIR, FTAS); 6.2; 6.3; PM funded EMD; PM funded Production/ESIP; Other (please identify)]

Multicore Considerations for Safety-Critical Software Applications — Title

N/A — Project

Brian Connell — Author/Project Engineer

ASEC-18Ø139 — Report number/Date received (to be completed by LCSD)

X1922 — Extension

B31/Rm23 — Building

RDAR-WSS-M-C — Author's Office Symbol

PART 1. **Must be signed before the report can be edited.**

a. The draft copy of this report has been reviewed for technical accuracy and is approved for editing.

b. Use Distribution Statement A X, B____, C____, D____, E____, or F____ for the reason checked on the continuation of this form. Reason: Content and wording are for broadest distribution

1. If Statement A is selected, the report will be released to the National Technical Information Service (NTIS) for sale to the general public. Only unclassified reports whose distribution is not limited or controlled in any way are released to NTIS.

2. If Statement B, C, D, E, or F is selected, the report will be released to the Defense Technical Information Center (DTIC) which will limit distribution according to the conditions indicated in the statement.

c. The distribution list for this report has been reviewed for accuracy and completeness.

Christopher Swanson 2/10/2021
Division Chief (Date)

PART 2. To be signed either when draft report is submitted or after review of reproduction copy.

This report is approved for publication.

Judy Mazeski 10 Oct 2018
Division Chief (Date)

RDAR-CIS (Date)

LCSD 49 (1 Sept 16)
Supersedes SMCAR Form 49, 20 Dec 06