Technische Universität Braunschweig

Institut für Datentechnik und Kommunikationsnetze

Error Handling within a Dual Core Lockstep RISC-V Processor Architecture

Author(s):

Martin Moya - m.moya@tu-braunschweig.de

Supervisor(s):

Alexander Dörflinger - adoerflinger@ida.ing.tu-bs.de

Examiner(s):

Prof. Dr.-Ing. Rolf ernst - ernst@ida.ing.tu-bs.de

Prof. Dr.-Ing. Harald Michalik - michalik@ida.ing.tu-bs.de

Abstract

Microcontrollers (MCU) are widely used in critical applications due to low-energy consumption and high-performance computing power. Despite these advantages, MCUs are sensitive to radiation like any other electronic device, leading to transient and interminent faults causing cathastrophic situations.

Critical applications have to function in a proper manner and deliver high level of Quality of Service (QOS), on the other hand, these kind of applications have also strict time and cost constrains, which means that they do not only have to meet high QOS standards, they also have to satisfy with a handfull of constraints. This work analyzes and proposes the development of a software solution for error handling within a Dual Core Lockstep (DCLS) RISC-V Processor Architecture. The solution provides a framework to implement different error handling techniques given specific scenarios in order to satisfy both requirements.

Contents

Chapter 1

Introduction

A system is considered *Safety-critical* when a failure in such could result in loss of life, significant property damage, or damage to the environment. Aircrafts, cars, weapons systems, medical devices and nuclear plants are considered traiditional examples of safety-critical systems. Most of these applications, if not all, rely on embedded systems that are expected to be fault tolerant. A system fault tolerant, according to the authors in **??**, is a system that is able to continue operating without interruption when one or more of its components fail, they aim to provide the ability to deliver a service that can be trusted, while fault removal and fault forecasting aim to reach confidence in that ability by justifying that the functional and the dependability and security specifications are adequate and that the systems is likely to meet them.

The objective of creating a fault-tolerant system is to prevent disruptions arising from a single point of failure, ensuring the high availability and business continuity of mission-critical applications or systems. As of now faults in distributed embedded systems can be permanent, intermittent or transient. Permanent faults cause long-term malfunctioning of components, while transient and intermittent faults appear for a short time. The effects of these faults, independently of their nature, can be devastating. They may corrupt data or lead to logic miscalculations, which can result in a fatal failure or dramatic QOS deterioration if not handled propperly.

Transient and intermittent faults can be addressed in *hardware* with hardening techniques or in *software*. Safety-critical applications have to be implemented such that they satisfy strict timing requirements and tolerate faults without exceeding a given amount of resources. Moreover, not only timeliness, reliability and cost-related requirements have to be considered but also other issues such as debugability and testability have to be taken into account.

In this chapter, we motivate our work on analyzing and implementing error handling software techniques for

Abreviaturas

DCLS Dual core Lockstep. 1, 3

MCU Microcontroller Unit. 1, 3

QOS Quality of Service. 1, 2, 3