Timing Analysis for Mode Switch in Component-based Multi-mode Systems

Yin Hang, Hans Hansson
Mälardalen Real-Time Research Centre (MRTC), Mälardalen University (MDH), Västerås, Sweden
Contact: young.hang.yin@mdh.se
Overview

- Introduction
- The Mode Switch Logic (MSL)
- The handling of atomic component execution
- The mode switch timing analysis
- Calculating the worst-case atomic component execution time
- Conclusions and future work
Introduction

- Growing complexity of embedded systems
- A promising design paradigm: Component-Based Software Engineering (CBSE)

Component reuse

- Multi-mode systems:
  - Distinguished behaviors in different operational modes
  - To reduce complexity and improve efficiency
  - E.g. the control software of an airplane
Introduction

● Our target:

● Component-Based Software Engineering (CBSE)

  +

● Multi-mode system

  II

● Component-Based Multi-Mode System (CBMMS)
Transformers: Optimus Prime

Human mode

Car mode
Introduction

• Component-based multi-mode system (CBMMS)

Primitive components: a, c, d, e, f
Composite components: Top, b

Composable mode switch?  Mode Switch Logic (MSL)
Introduction

- Atomic component execution--Atomic Execution Group (AEG)

![Diagram of Atomic Execution Groups (AEGs)]
Overview

- Introduction
- The Mode Switch Logic (MSL)
- The handling of atomic component execution
- The mode switch timing analysis
- Calculating the worst-case atomic component execution time
- Conclusions and future work
The Mode Switch Logic (MSL)

- Mode-aware component model
- Mode mapping mechanism
- Mode Switch runtime mechanism
Mode Switch Logic (MSL)

- The Mode Switch (MS) propagation mechanism

MSR: Mode Switch Request
MSI: Mode Switch Instruction
MSS: Mode Switch Source
MSDM: Mode Switch Decision Maker
The mode switch runtime mechanism

- Guaranteeing mode consistency—**Mode switch dependency rule**
  - A component starts mode switch after receiving or actively issuing an MSI.
  - As a component starts its mode switch, if its target mode is different from its current mode, it will do the reconfiguration.
  - A component who has received an MSI from its parent must send an **MSC (Mode Switch Completion)** back after it completes its mode switch.
  - Conditions for mode switch completion:
    - Primitive component: after reconfiguration
    - Composite component: (after reconfiguration) + the collection of all expected MSC from the subcomponents

The mode switch completion of a system = the mode switch completion of the MSDM.
The mode switch runtime mechanism

- The complete mode switch process
Overview

- Introduction
- The Mode Switch Logic (MSL)
- The handling of atomic component execution
- The mode switch timing analysis
- Deriving the worst-case atomic component execution time
- Conclusions and future work
The handling of atomic component execution

Mode: m1

AEGI

AEG: Atomic Execution Group
The handling of atomic component execution
Overview

● Introduction
● The Mode Switch Logic (MSL)
● The handling of atomic component execution
● The mode switch timing analysis
● Deriving the worst-case atomic component execution time
● Conclusions and future work
The mode switch timing analysis without AEG

- **Notations**
  - $t_{MSR}$, $t_{MSI}$, $t_{MSC}$: The transmission time of an MSR, MSI or MSC
  - $RCT_{ci}$: The reconfiguration time of $c_i$
  - $MS_{ci}$: The mode switch time of $c_i$

- **Two phases**
  - $T_{MSR}$: MSR propagation (upstream)
  - $T_{MSI}$: MSI propagation (downstream) and mode switch
The mode switch timing analysis without AEG

- The first phase--$T_{MSR}$
- Depth level: L
- MSR propagation time

$$T_{MSR} = t_{MSR} \times \Delta L$$

$$\Delta L = L_{MSS} - L_{MSDM}$$

MSS: Mode Switch Source
MSDM: Mode Switch Decision Maker
The mode switch timing analysis without AEG

- The second phase -- $T_{MSI}$

\[ T_{MSI} = MS_{MSDM} \]

Total mode switch time:
\[ MS = T_{MSR} + T_{MSI} \]

For a primitive component $x$: \[ MS_x = RCT_x \]
A constant delay, the worst-case execution time of an AEG (AE), is added to the reconfiguration starting time of the AEG component and its activated subcomponents

\[ MS_b = \max \{ RCT_b + AE_b, t_{MSI} + MS_d + t_{MSC} + AE_b, t_{MSI} + MS_e + t_{MSC} + AE_b, t_{MSI} + MS_f + t_{MSC} \} \]
Overview

- Introduction
- The Mode Switch Logic (MSL)
- The handling of atomic component execution
- The mode switch timing analysis
- Calculating the worst-case atomic component execution time
- Conclusions and future work
Calculating the worst-case atomic component execution time—AE

- A model-checking approach to deriving AE--UPPAAL

- Parameters:
  - Input data rate: R
  - The data processing time of each primitive component: C
  - Maximum of data elements in the AEG: N
Calculating the worst-case atomic component execution time—AE
Calculating the worst-case atomic component execution time—AE

- The “sup” operator: automatically returning the maximum value of a variable or clock

AE: the maximum value of Clock z
Verification

Returning the maximum buffer usage

Returning the worst-case scenario

Returning AE
Conclusions and future work

● Mode Switch Logic (MSL) for component-based multi-mode systems (CBMMSs)
● The handling of atomic component execution in the MSL
● The mode switch timing analysis
● Deriving the worst-case atomic component execution time by model checking

Future work

● Resolving the conflict of multiple mode switch triggering+the mode switch timing analysis
● More general model for the Atomic Execution Group (AEG)
● Component connections with feedback loops
Thank you!