HITACHI
Inspire the Next

2nd Workshop on Certifiable Multicore Avionics and Automotive Systems

Multicore Migration Study in Automotive Powertrain Domain

21/Apr/2017
Hitachi, Ltd.,
Takeshi Fukuda、Tasuku Ishigooka, Fumio Narisawa
Hitachi Automotive Systems, Ltd.,
Toru Irie, Takafumi Suzuki

Contents

1. Outline of Hitachi
2. Background
3. Control Software and Issue
4. Our Method
5. Evaluation with Engine Control Software
6. Summary and Discussion

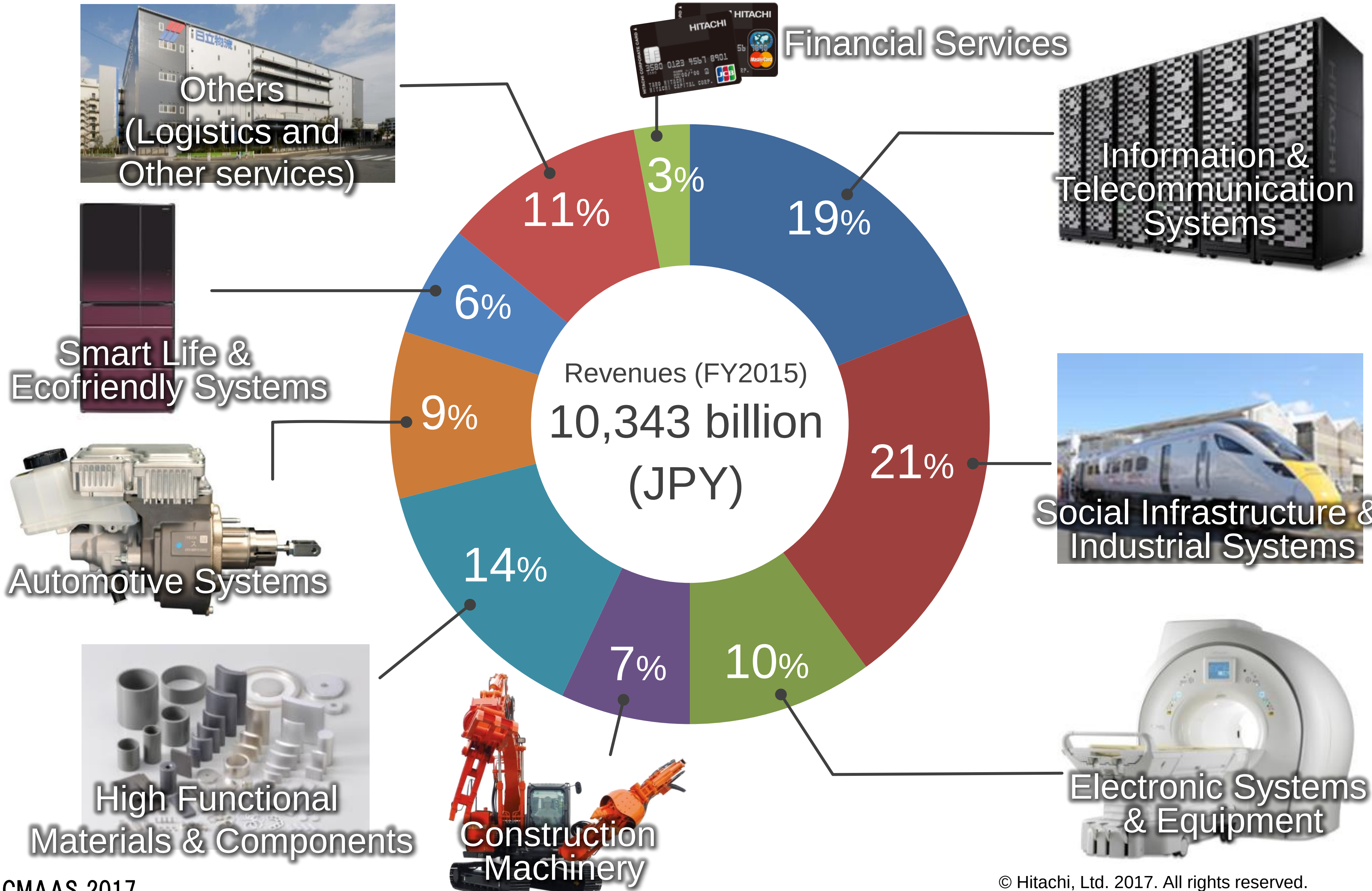
Hitachi corporate data

HITACHI
Inspire the Next

Hitachi, Ltd.	
Incorporated	1st February 1920 [Founded 1910]
Head Office	Marunouchi 1-6-6, Chiyoda-ku, Tokyo, JAPAN

Hitachi Group (Consolidated) FY2015	
Revenues	JPY 10,343 billion
R&D expenditure	JPY 333.7 billion [3.3% of revenue]
No. of subsidiaries	1,056 companies [262 domestic; 794 overseas]
No. of employees	335,244 [187,936 domestic; 147,308 overseas]

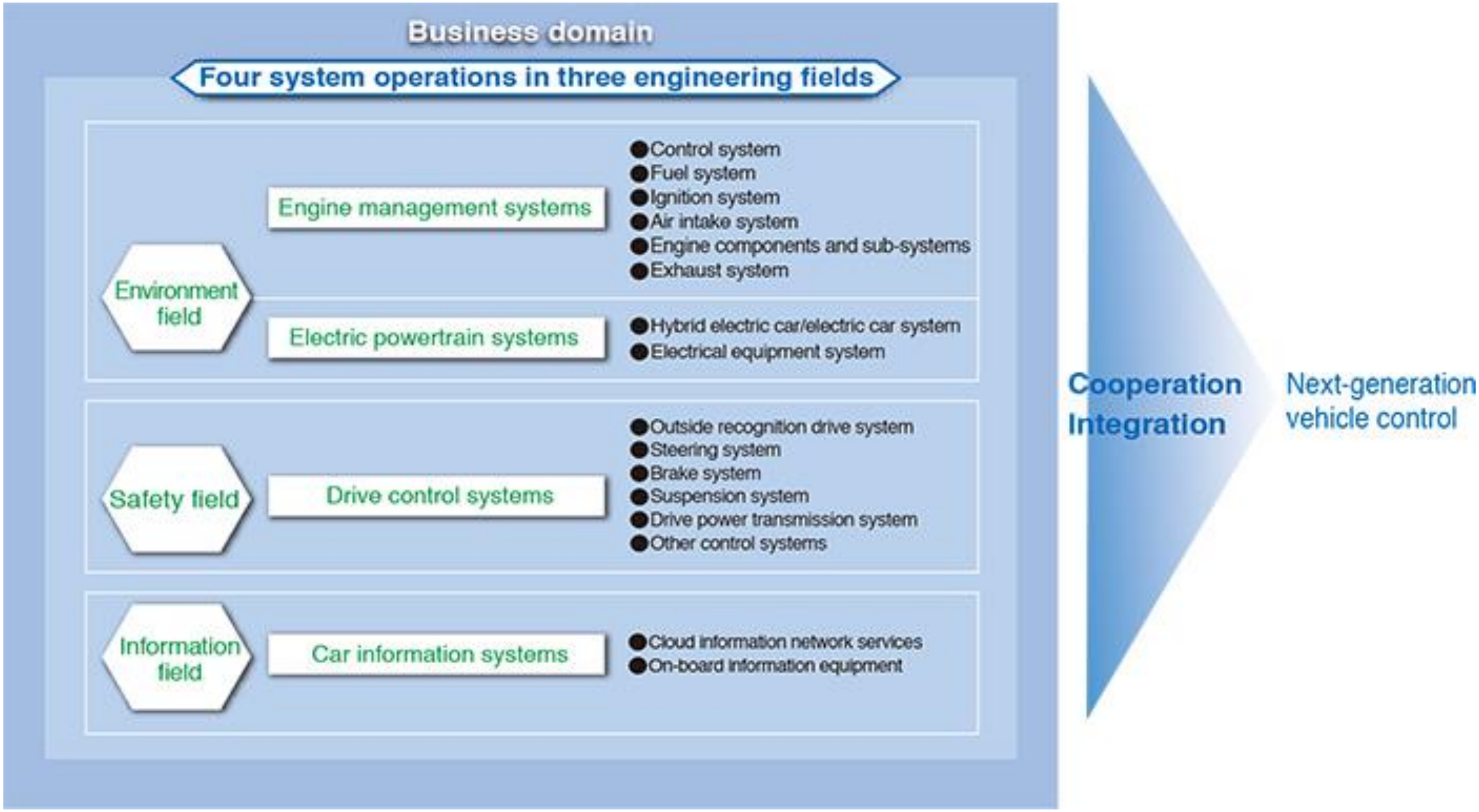
Business portfolio



Business Description of Automotive Dep.

HITACHI
Inspire the Next

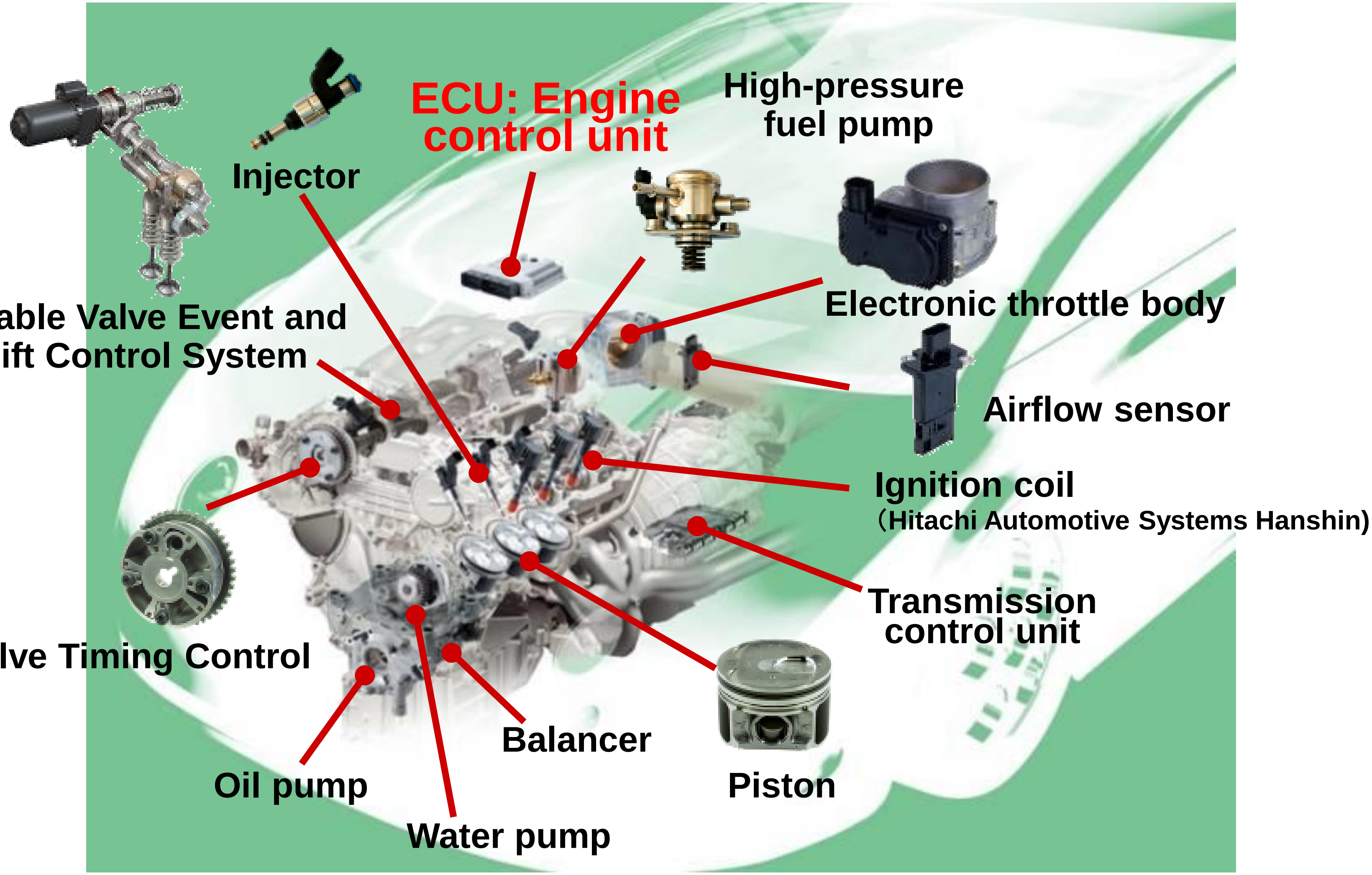
System solutions that Hitachi Automotive Systems will provide for progress in “Environment”, ”Safety” and ”Information” field.



[Environment] Engine Management Systems

HITACHI
Inspire the Next

Offering environment-friendly and high-efficiency engine management systems for reduction in CO_2 and prevention of air pollution.



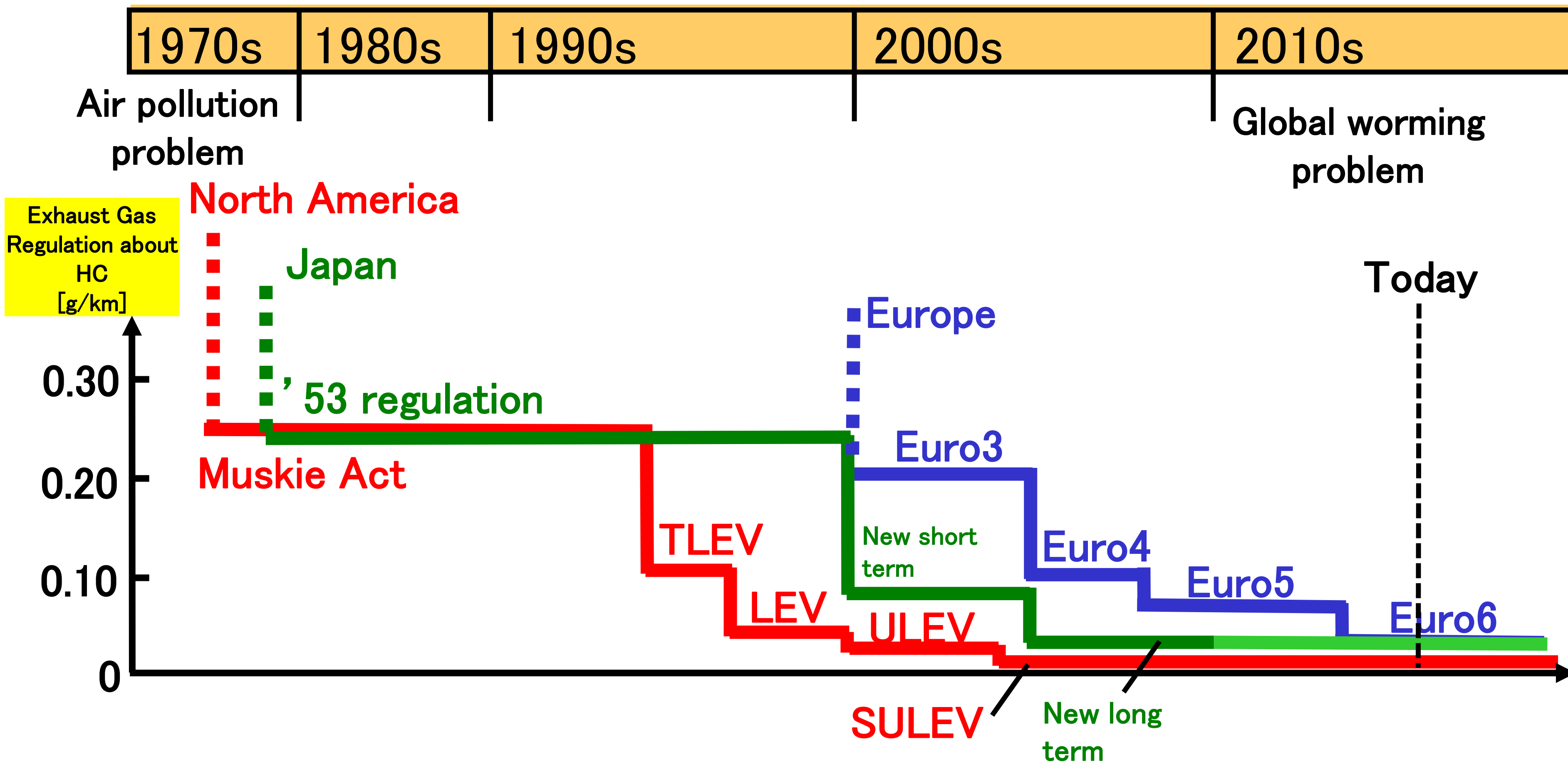
Contents

1. Outline of Hitachi
2. **Background**
3. Control Software and Issue
4. Our Method
5. Evaluation with Engine Control Software
6. Summary and Discussion

Background: Trend of Exhaust Gas Regulation

HITACHI
Inspire the Next

The performance requirements of automotive engine control are increasing, for instance to comply the exhaust emission regulations and reduce gasoline consumption.

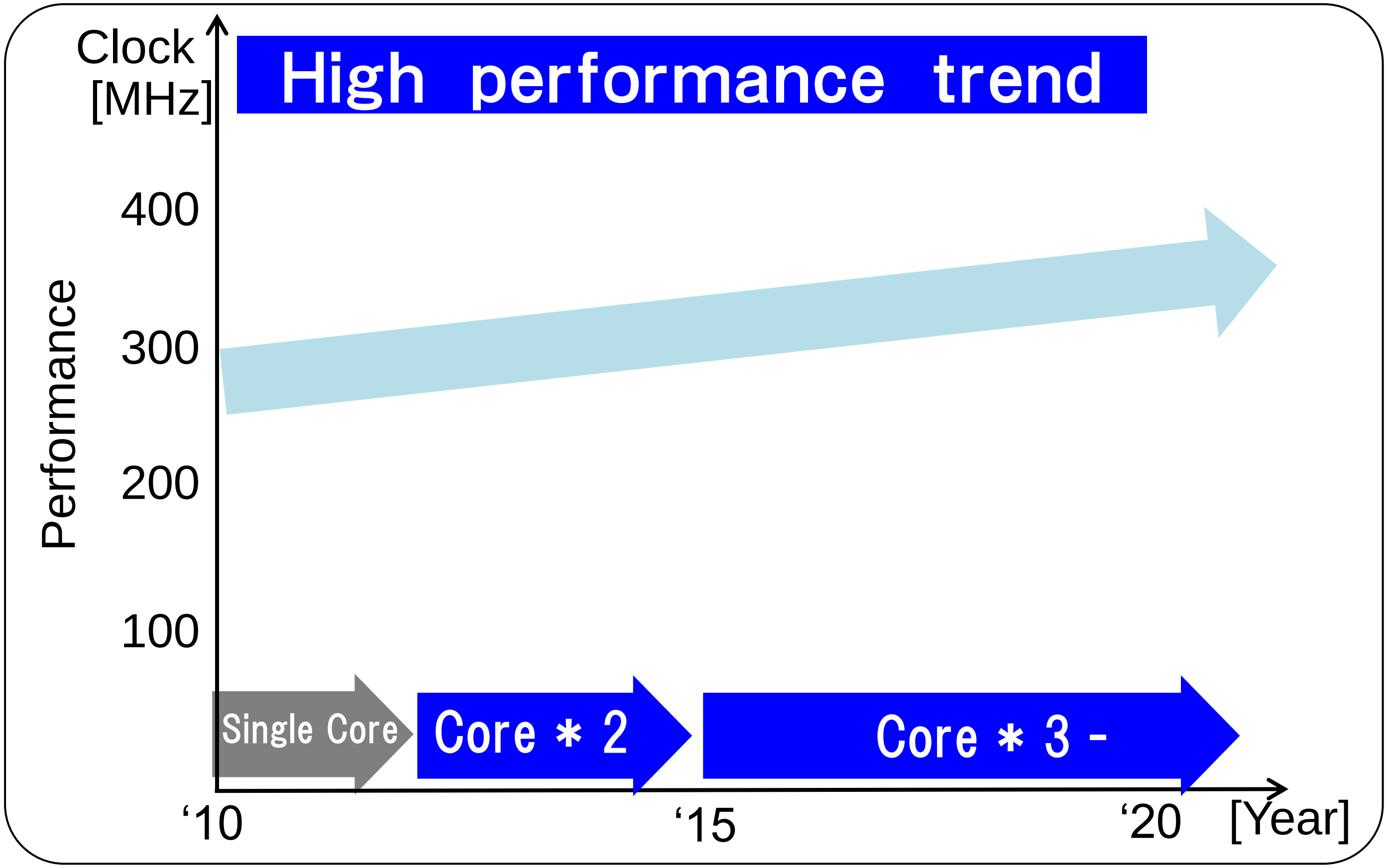


Trend of Microcontroller for Powertrain

HITACHI
Inspire the Next

Current electronic controller unit requests high performance microcontroller.

■Trend of Microcontroller for Powertrain Systems



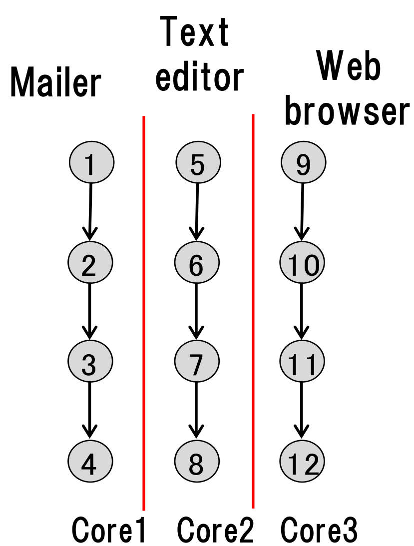
ECU: Electric Control Unit

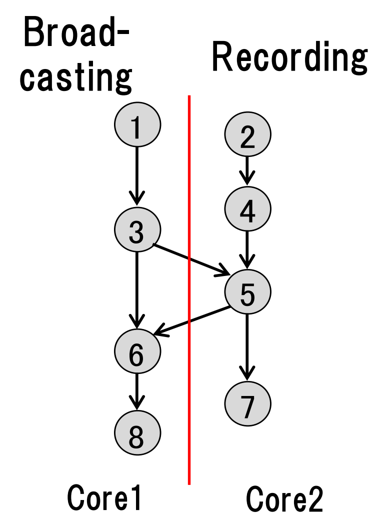
HITACHI
Inspire the Next

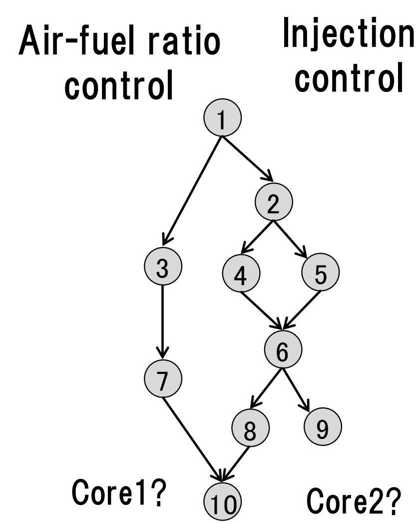
Our study: To migrate legacy source code to multicore platform.

■Comparison with other products

	PC, Smart phone	Television	Automotive engine
Data Dependency	Loosely coupled	Loosely coupled	Complicated







① Task + Execution order

→ Dataflow

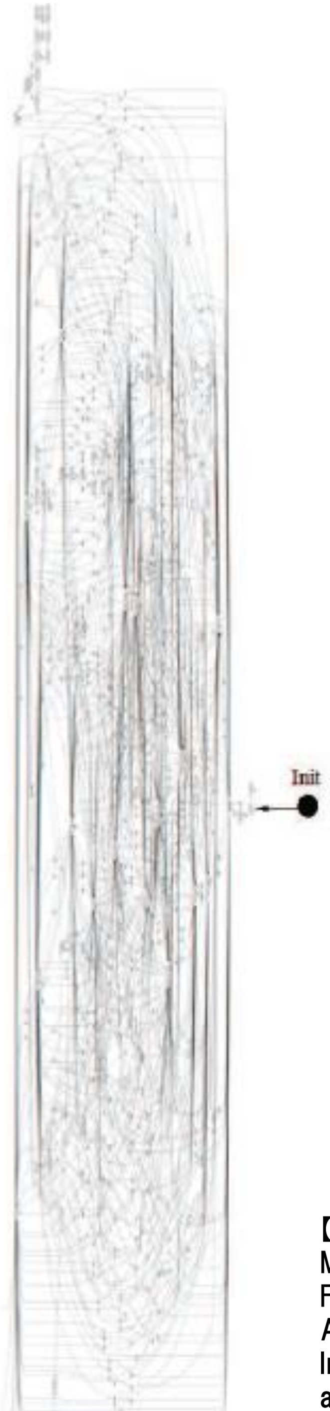
Our study

Our study ... atform.

■Comparison

... e engine

Data Depen ... cated



A potion of dataflow of
engine control software

Injection
control

① Task + Execut

→ Dataflo

2

5

6

9

Core2?

【Cite】
M. Panić, S. Kehr, E. Quiñones, B. Boddecker, J. Abella and F. J. Cazorla, RunPar: An Allocation Algorithm for Automotive Applications Exploiting Runnable Parallelism in Multicores, International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS'14), Oct. 2014.

HITACHI
Inspire the Next

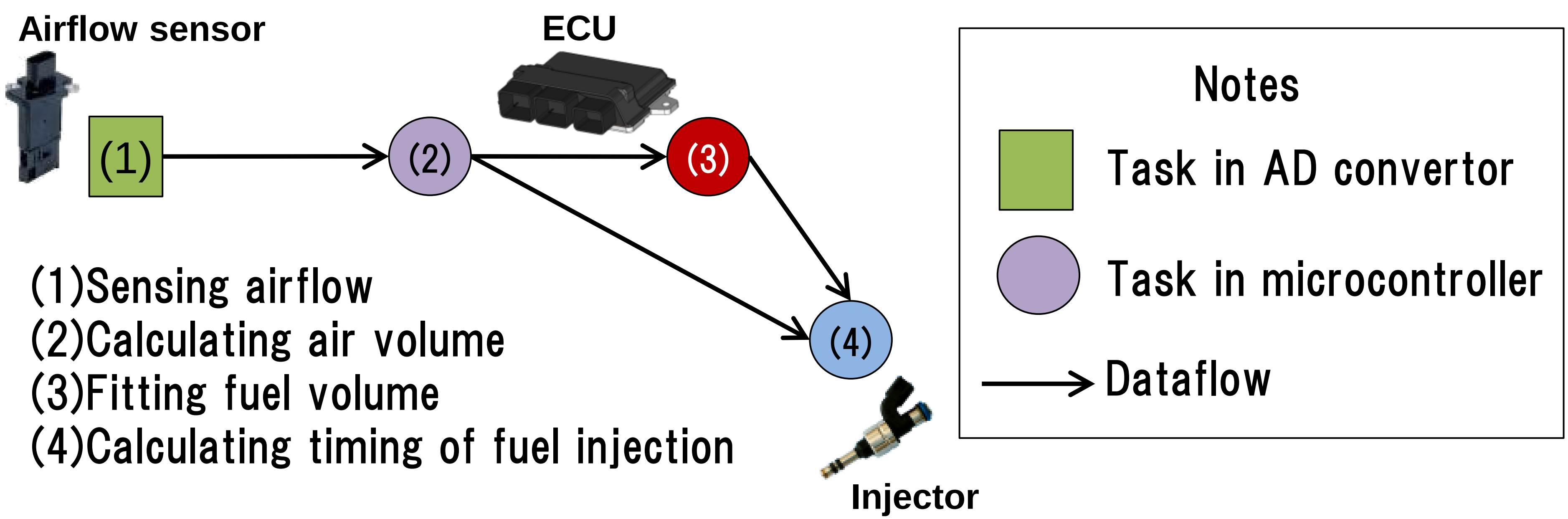
Contents

1. Outline of Hitachi
2. Background
3. **Control Software and Issue**
4. Our Method
5. Evaluation with Engine Control Software
6. Summary and Discussion

Dataflow of Control Software

Dataflow of control software roughly consists of sensing, calculating, and actuating.

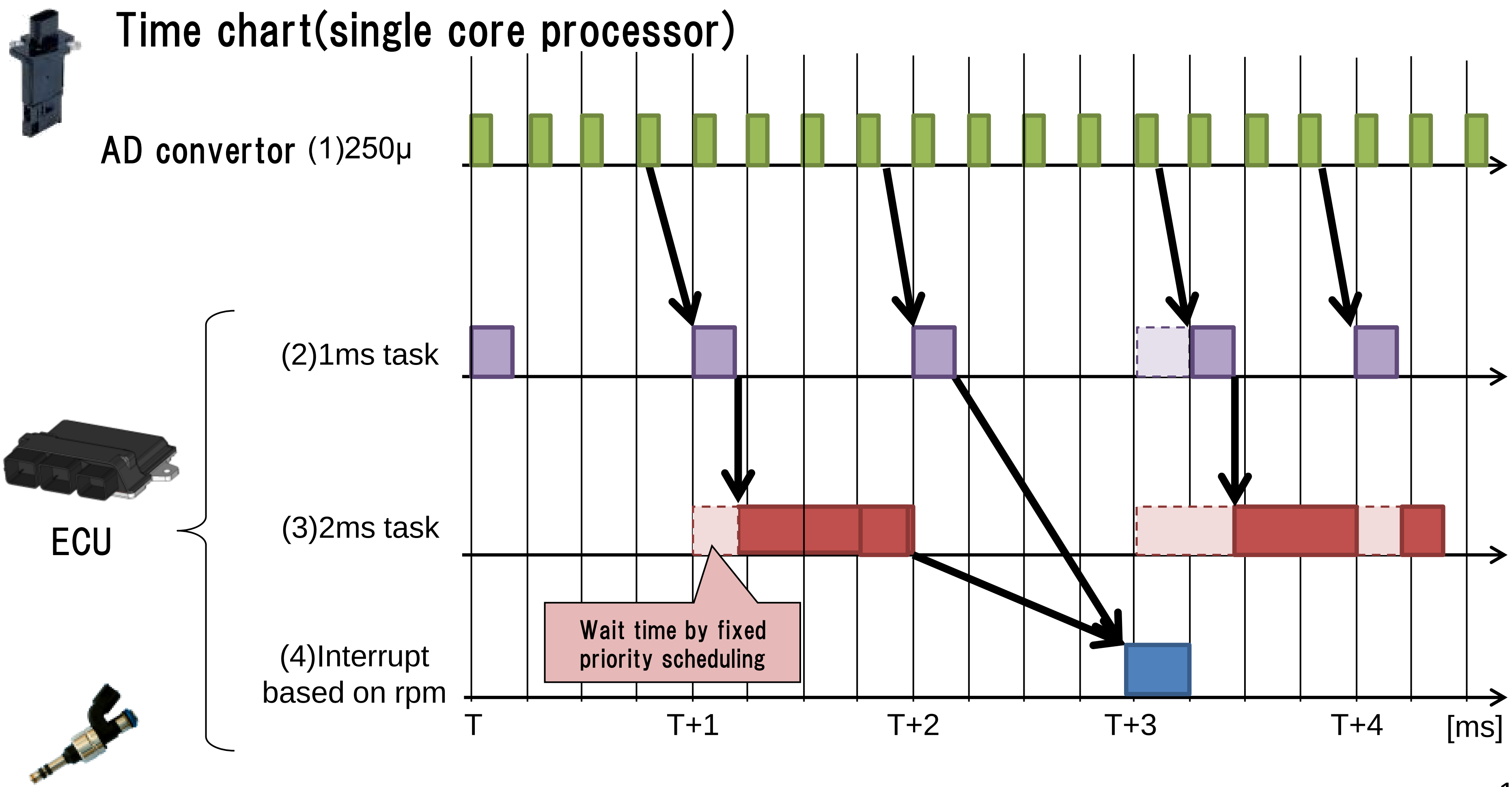
E.g. Dataflow of fuel injection



(1)Sensing airflow
(2)Calculating air volume
(3)Fitting fuel volume
(4)Calculating timing of fuel injection

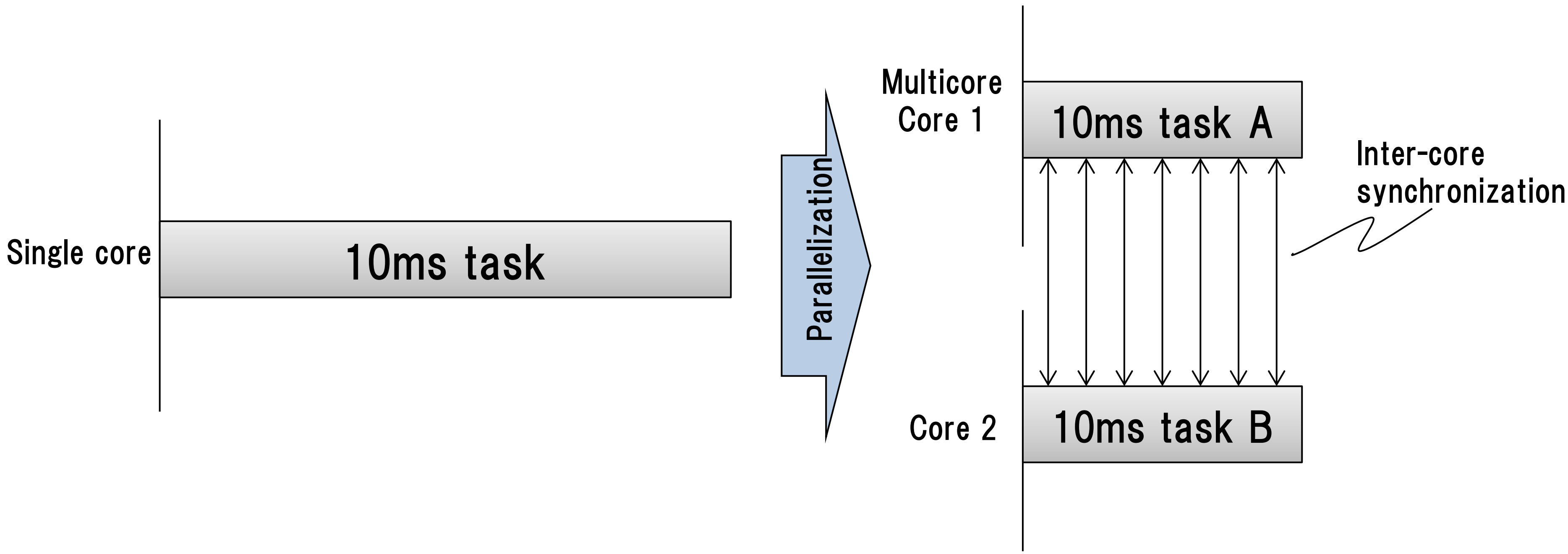
Cyclic Execution of Control Software

Tasks are executed periodically with each cyclic time or executed sporadically based on event like an engine rotation speed.



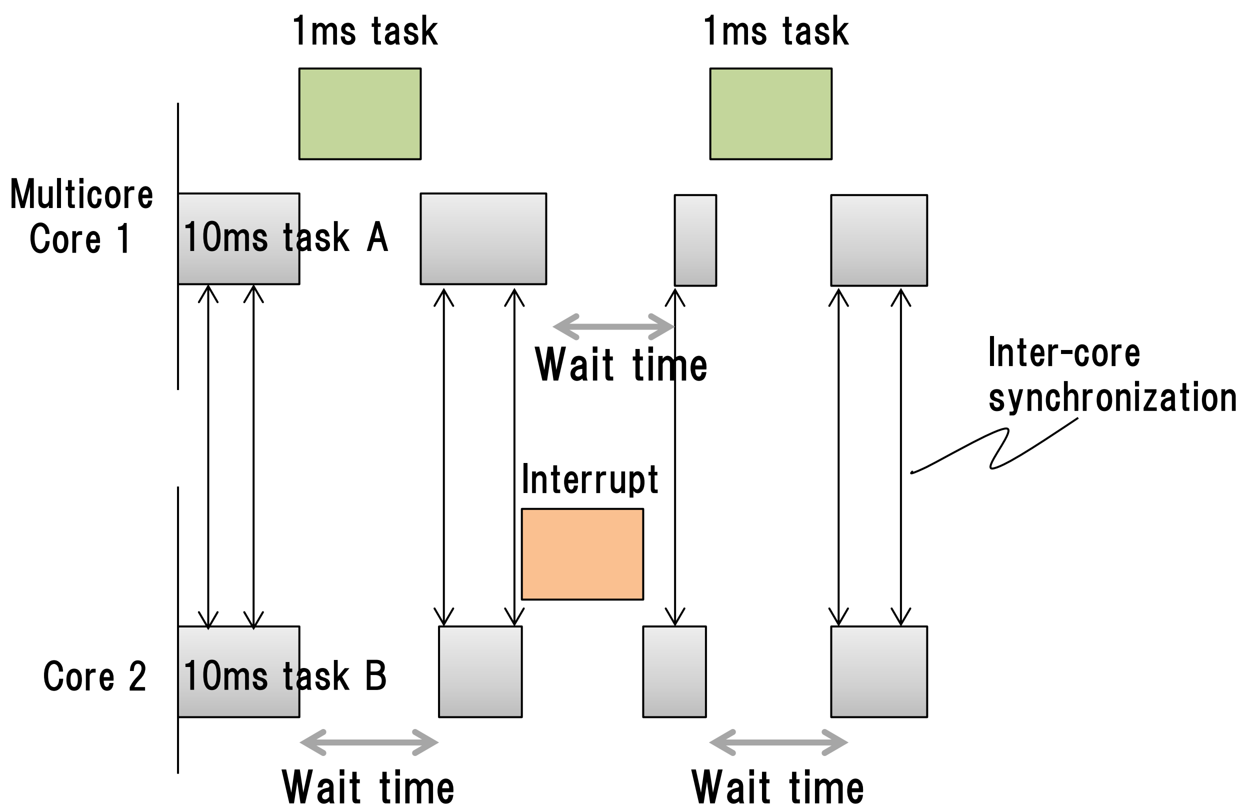
Issue of parallelization

A large number of inter-core communication data causes high frequent inter-core synchronization between cores.



Issue of parallelization（Cont’d)

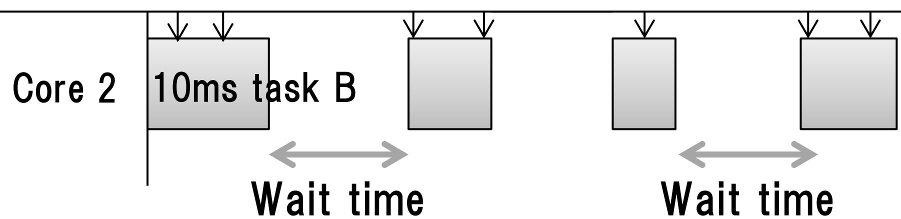
Issue: Long wait time is frequently happened when one core is interrupted by other high priority tasks due to frequent inter-core synchronization.



Issue of parallelization（Cont’d)

Issue: Long wait time is frequently happened when one core is interrupted by other high priority tasks due to frequent inter-core sync

【Our Goal 】
In order to achieve high parallelization, our goal is to reduce a number of inter-core synchronization.



Inspire the Next

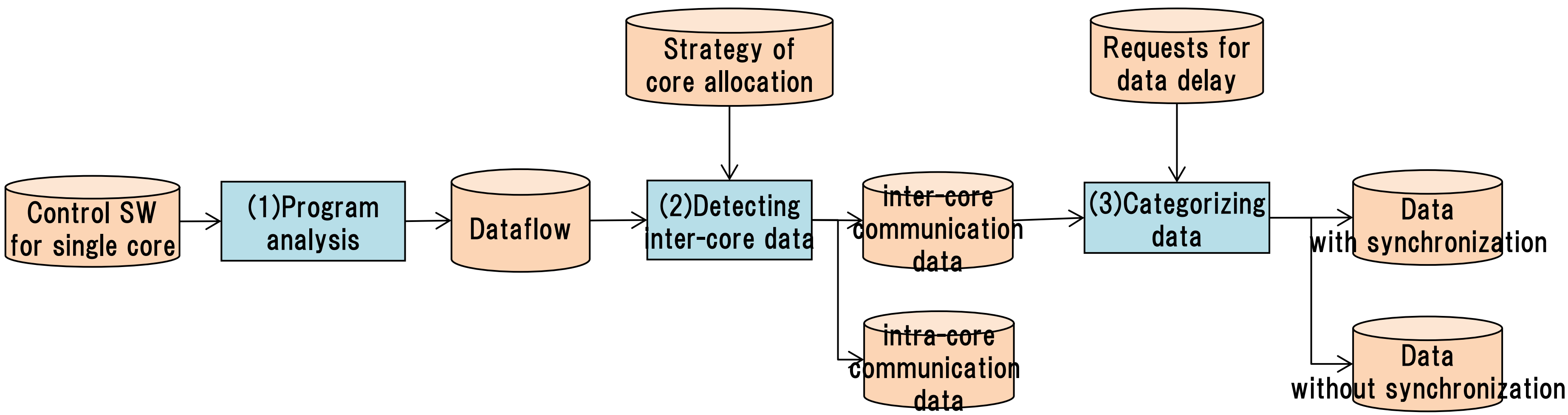
Contents

1. Outline of Hitachi
2. Background
3. Control Software and Issue
4. **Our Method**
5. Evaluation with Engine Control Software
6. Summary and Discussion

Overview of our method

HITACHI
Inspire the Next

We proposed parallelization method with performance requirements of system control.
Feature of our method is to identify the inter-core data which doesn't need inter-core synchronization according to the requirements of "Data Delay Time".

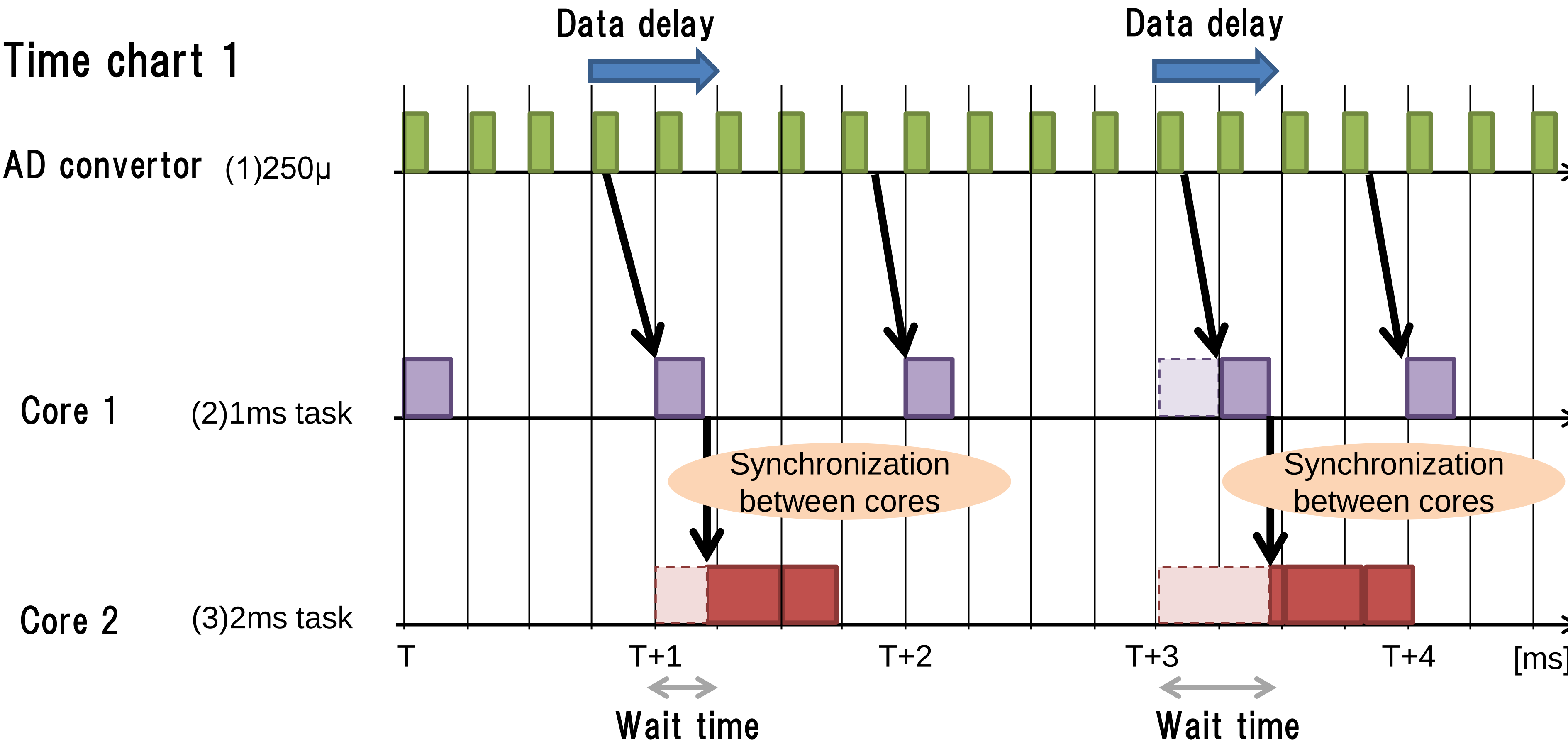


Data delay time

HITACHI Inspire the Next

Data delay time is one kind of end-to-end path latency, especially, from start time of sensing task to start time of other task which has data dependence with sensing task.

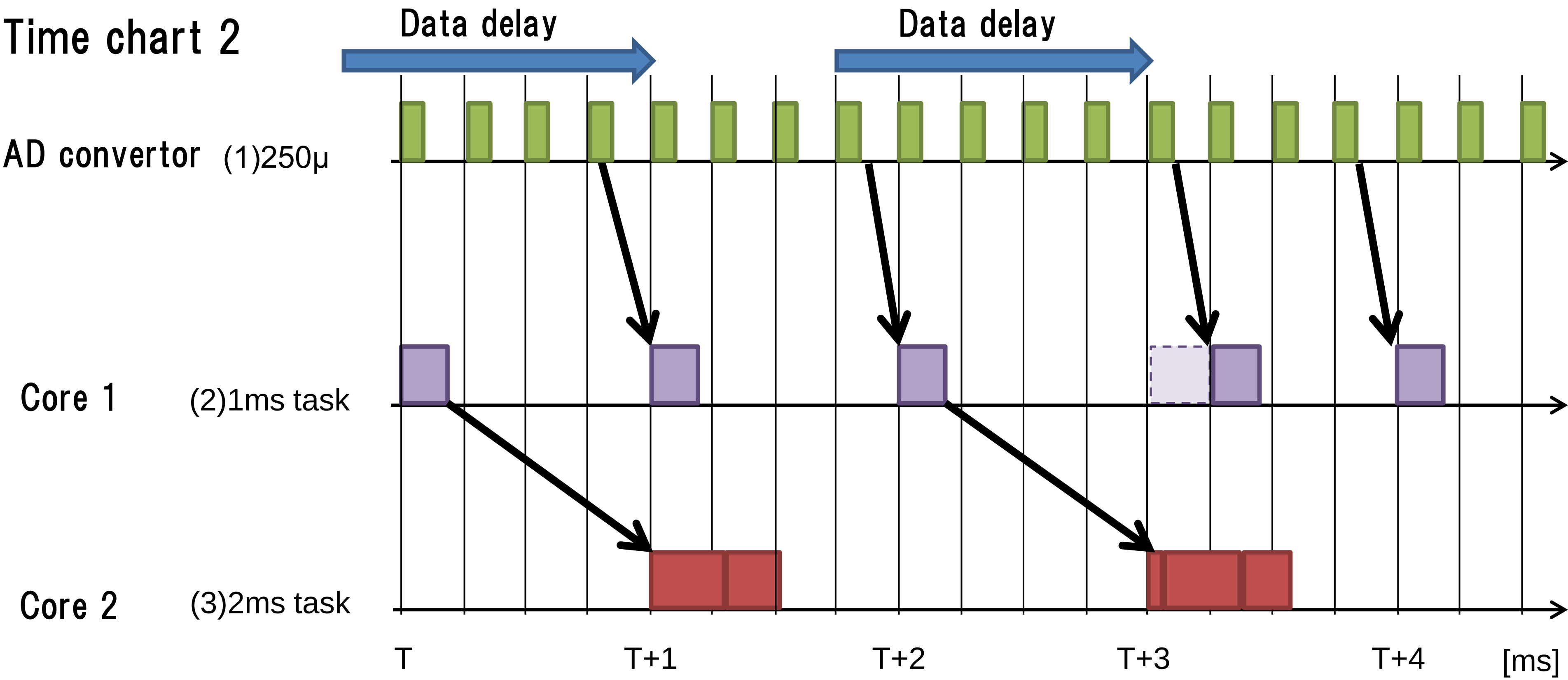
With synchronization between cores



Data delay time（cont’d)

Without synchronization between cores

Time chart 2



No wait time

Idea

HITACHI
Inspire the Next

Our idea: we select the parallelization option which is without synchronization between cores when we can allow a WCRT(Worst Case Response Time) of data delay time.

	With synchronization	**Without synchronization**
Input data	Same timing data	Previous task data
Wait time	Long	Zero
Parallelization	Low	High
Data delay time	Short	Long
WCRT of data delay time	Short	Long $\sum_{i=1}^{N-1}(R(\tau_i)+\hat{n}T(\tau_{i+1})-r(\tau_{i+1})+S(\tau_N))$ 【Cite】 Balsini, A., Melani, A., Buonocunto, P. and Natale, Ki, M.: FMTV 2016: Where is the Actual Challenge?, 3rd Challenge on Formal Method for Timing Verification (FMTV), International Workshop on Analyais Tools and Methodologies for Embedded and Real-Time Systems (WATERS 2016), (2016)

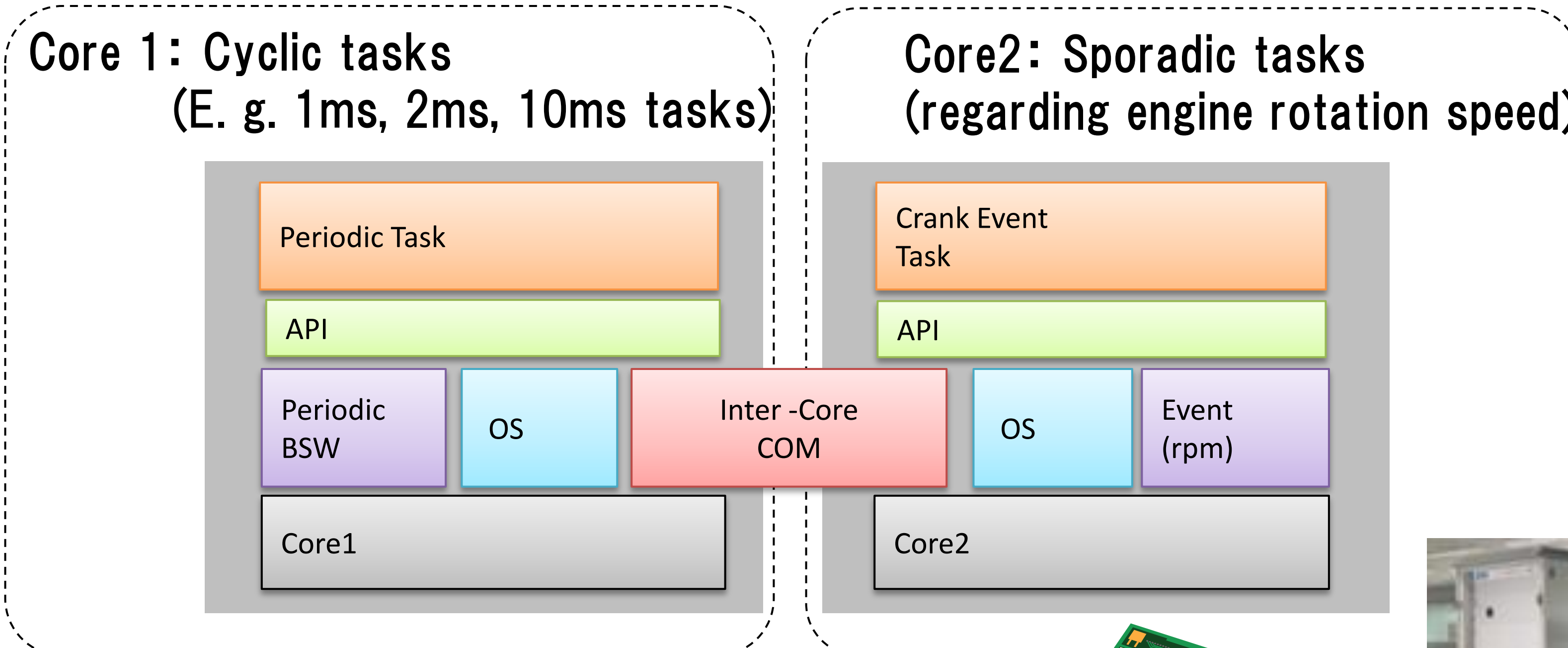
Contents

1. Outline of Hitachi
2. Background
3. Control Software and Issue
4. Our Method
5. **Evaluation with Engine Control Software**
6. Summary and Discussion

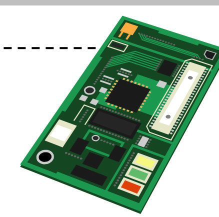
Overview of evaluation

HITACHI
Inspire the Next

We migrated whole legacy engine control software to multicore ECU which has two cores. And evaluated it with HILS.



【Engine control SW】included aprox. 10000 control signals
【ECU】 two cores



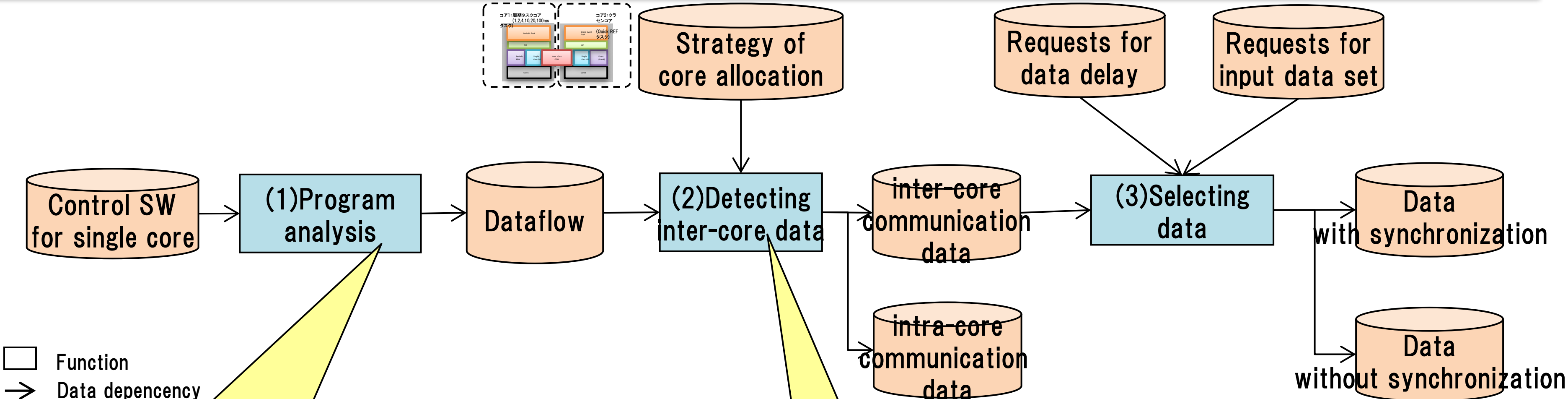
ECU
(Engine Controller Unit)

HILS
(Hardware-in-the-loop Simulation)

Total amount of Inter-core communication data

HITACHI
Inspire the Next

We found that a total amount of inter-core communication data is approx. 600 in our process step two.



Dataflow(portion)
the result of program analysis

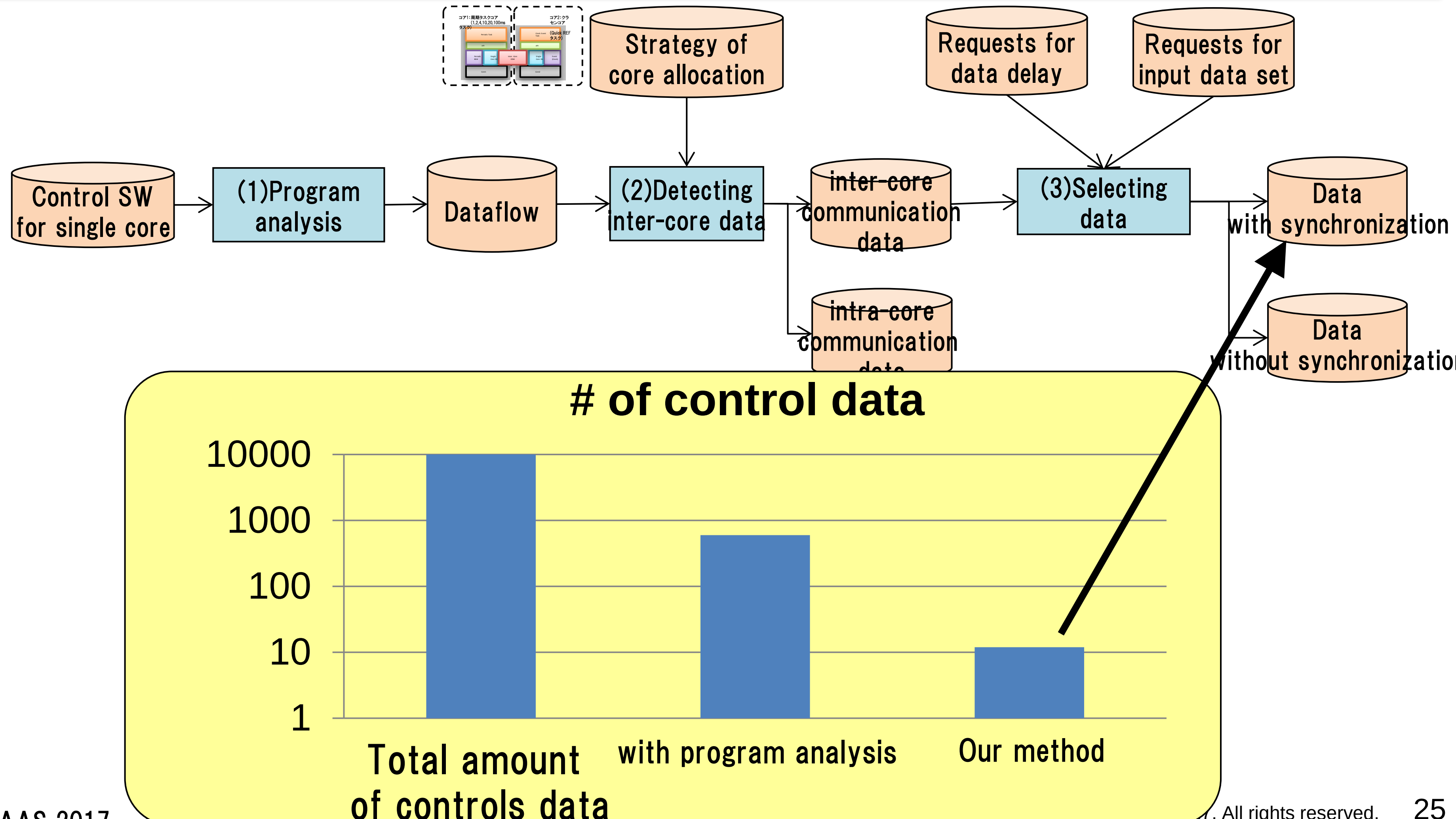
Variable / Type	CORE1	CORE2	コア間／コア内判定
variable_231	R/W	W	コア間
variable_232	R	R/W	コア内
variable_233	R/W	R	コア間
variable_234	−	2	コア内
variable_235	R	R/W	コア間
variable_236	R	R/W	コア間
variable_237	−	R/W	コア内
variable_238	R/W	W	コア間
variable_239	R/W	W	コア間
variable_240	R/W	R	コア間
variable_241	R/W	R	コア間
variable_242	W	1	コア間
variable_243	−	R/W	コア内
variable_244	−	R/W	コア内
variable_245	−	R/W	コア内

List of Inter-core communication data
the result of detecting inter-core data

The results

HITACHI
Inspire the Next

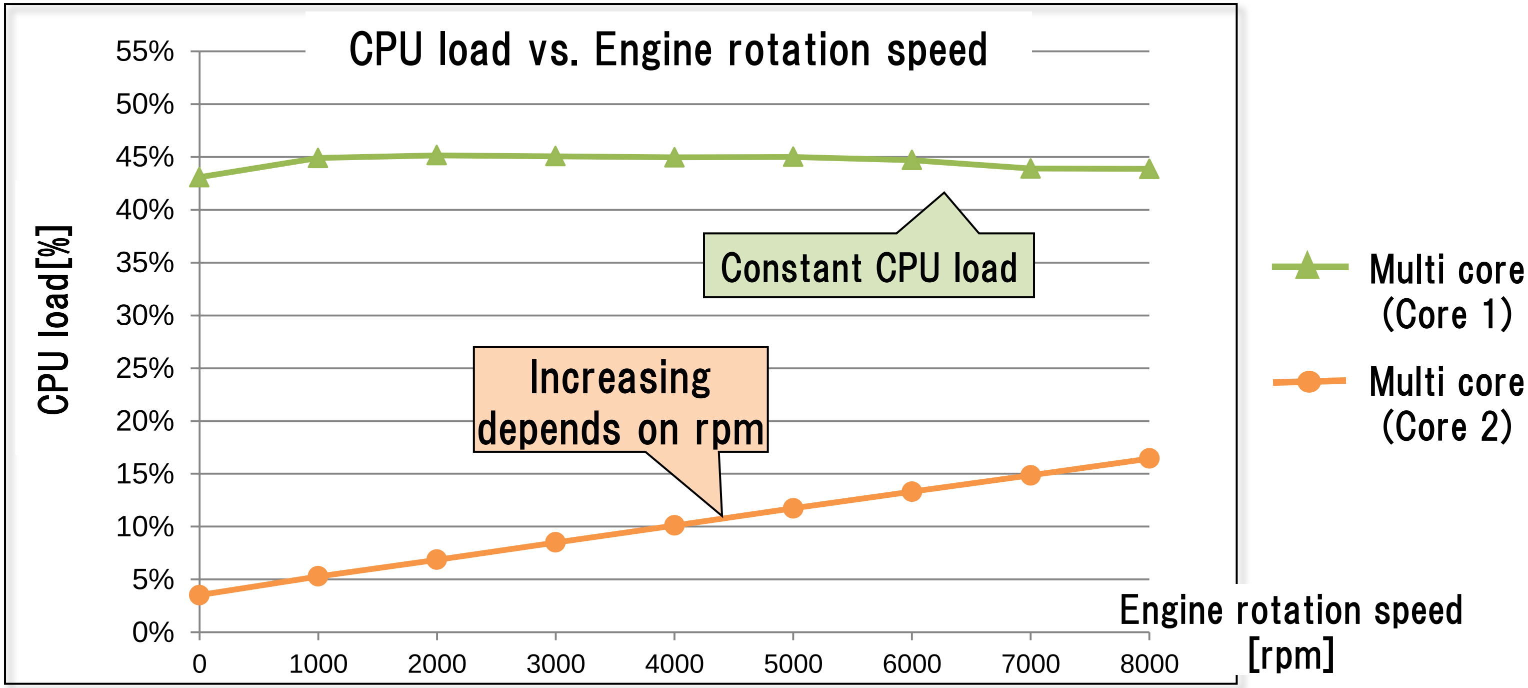
Our method indicated that more than 90% out of approx. 600 inter- core communication data don’t need synchronization.



Evaluation with HILS (Hardware-in-the-loop Simulator)

HITACHI
Inspire the Next

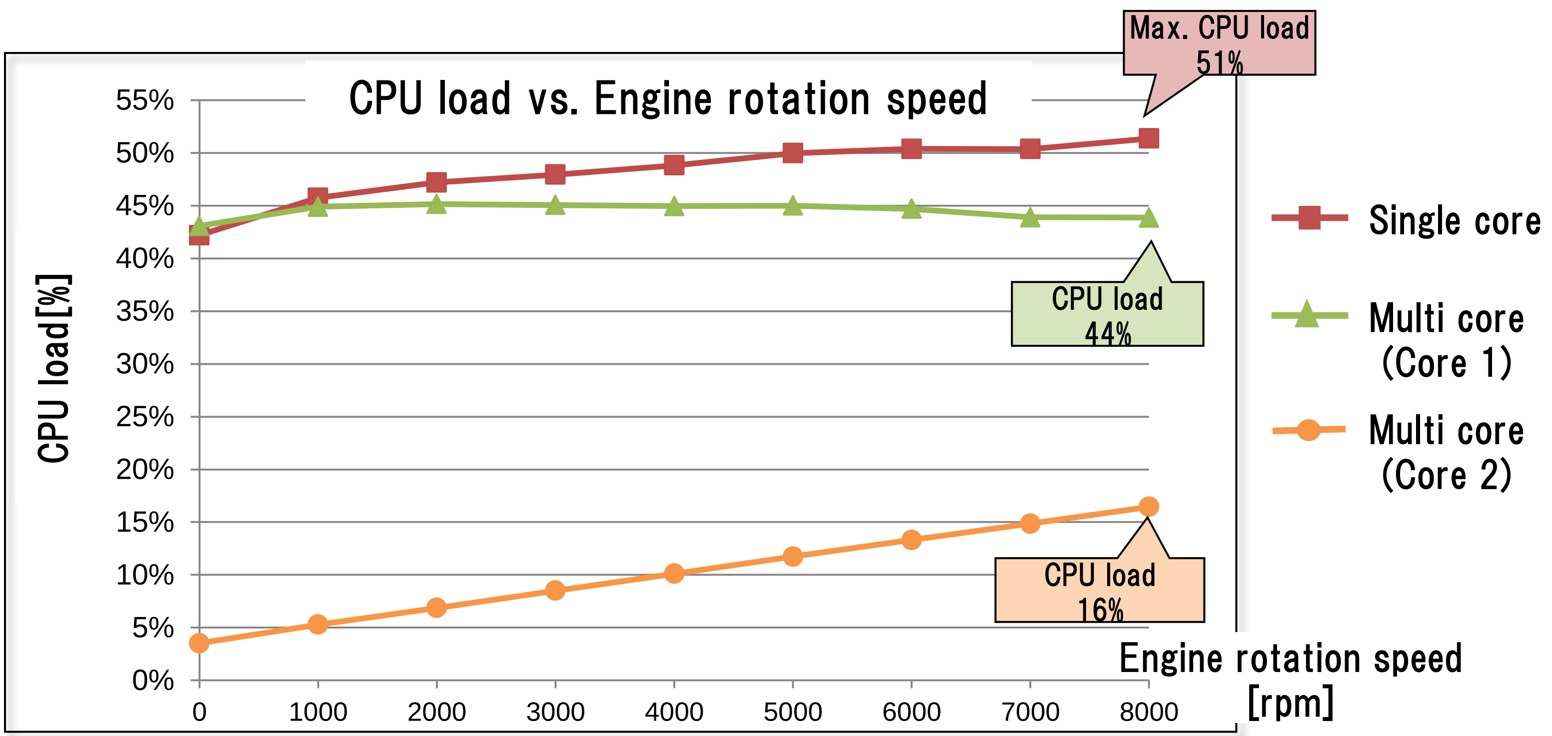
Result of HILS evaluation indicated that our method is
(1)useful for parallelizing engine control software
(2)
(3)



Evaluation with HILS (Hardware-in-the-loop Simulator)

HITACHI
Inspire the Next

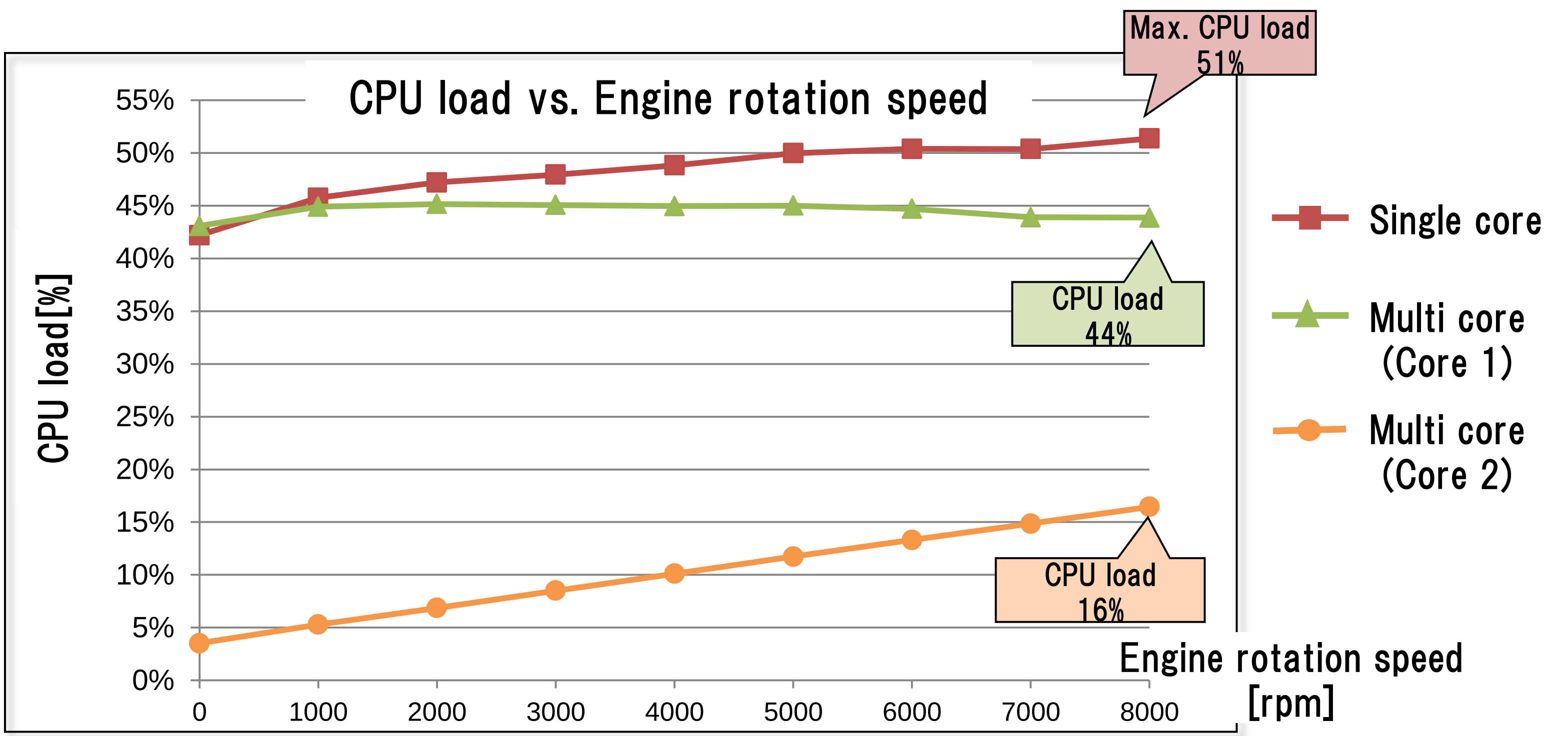
Result of HILS evaluation indicated that our method is
(1)useful for parallelizing engine control software
(2)able to distribute CPU load to cores
(3)



Evaluation with HILS (Hardware-in-the-loop Simulator)

HITACHI
Inspire the Next

Result of HILS evaluation indicated that our method is
  (1)useful for parallelizing engine control software
  (2)able to distribute CPU load to cores
  (3)Keeping hard real-time deadline. e.g. fuel infection



Contents

1. Outline of Hitachi
2. Background
3. Control Software and Issue
4. Our Method
5. Evaluation with Engine Control Software
6. **Summary and Discussion**

Summary

【Proposed method】
・Parallelization method for control software
・The feature of our method is
  -identify the inter-core data which doesn't need core synchronization according to the requirement of "Data Delay Time"

【Evaluation】
・We apply our method to whole legacy engine control software.

【Results】
・More than 90% of data out of approximately 600 inter-core communication data don't need synchronization mechanism.

・The evaluation results with HILS indicated that the parallelized software satisfied requirements of real-time performance.

HITACHI
Inspire the Next

【 Discussion 】

・Process and Method to decide requirement time of data delay with control engineer is required.

It is tough issue to decide it which doesn’t cause a bad effect for the performance of systems.

It is a nonsense if multicore migration decrease engine performance like fuel efficiency.

・Precise WCRT analysis method is required.

The result of WCRT analysis for data delay time is too pessimistic.

Thank you!