

# Hot Trace Detector: A Support Architecture for APC

Hanpei KOIKE<sup>1,2</sup>, Tomonori HIRUTA<sup>3</sup>, Hayato YAMANA<sup>1,3</sup>  
(<sup>1</sup> APC Technology Group <sup>2</sup> National Institute of AIST <sup>3</sup> Waseda University)

## Contents

- Motivation
- Trace Extraction Mechanism
- Hot Trace Detector
- Reconfigurable Experimental Machine REX
- Trace Extraction Experiment
- Conclusions

2

## Motivation

- Speculative Execution: Optimization method which exploits **Statistical Nature** of the program execution
- It is essential to know the statistical nature of the program and to feed back the information to the compiler
- What is the architecture which supports *Advanced Parallelizing Compiler* effectively?

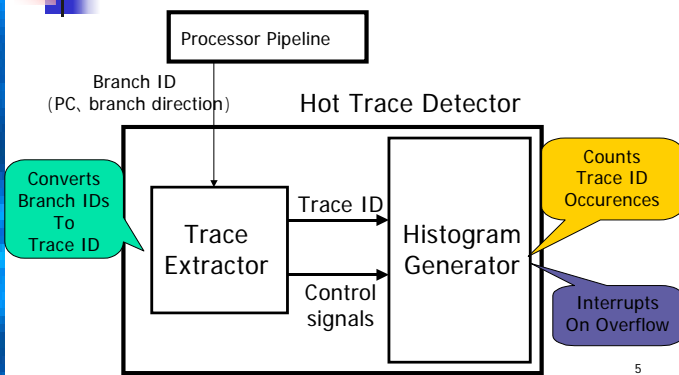
3

## Information Extracting Mechanism for Compilers

- Target program characteristics in this study
  - Traces**(Control Flow among Multiple Branches)
- Hot Trace Detector
  - Hardware Support for hot trace extraction
    - H.T.D. creates histogram of frequent traces consisting of series of multiple branches
  - Cf.: Hot Spot Detector (Illinois)
    - H.S.D. creates only histogram of each *single* branch

4

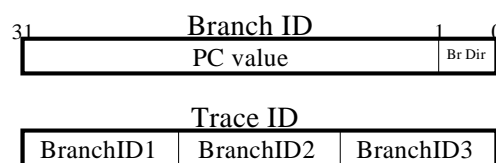
## Hot Trace Detector



5

## Branch ID and Trace ID

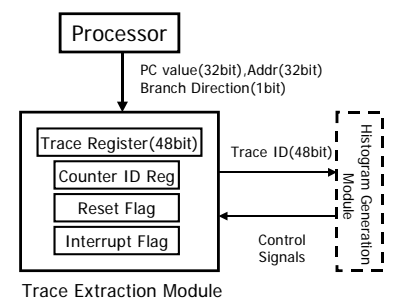
- Branch ID
  - Branch ID = PC value + Branch Direction
- Trace ID
  - Trace ID is a series of Branch IDs



6

## Trace Extraction Module

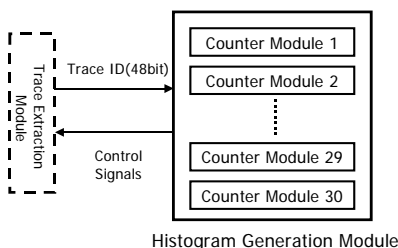
- Combines Branch IDs to Convert to Trace ID



7

## Histogram Generation Module

- Receives Trace IDs and Generates Histogram



8

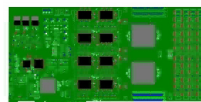
## REX: Reconfigurable Experimental Machine

- We Develop Hot Trace Detector on our FPGA based experimental machine: REX
- REX Allows Efficient Architectural Experiments.

With MIPS processor on REX,  
SPECint95 129.compress runs  
**7.5** times faster than SimpleScalar

Compress execution time	
REX	SimpleScalar *
155 min.	1160 min.

\* ver3.0 Sim-bpred



9

## Trace Extraction Experiment

- We designed Hot Trace Detector in Verilog HDL and connected it to the MIPS processor core running on the REX
- SPECint95 129.compress (test data set) was executed to collect trace information
- Compress trace diagram was drawn from the extracted information
- Purpose: To demonstrate the function of the Hot Trace Detector

10

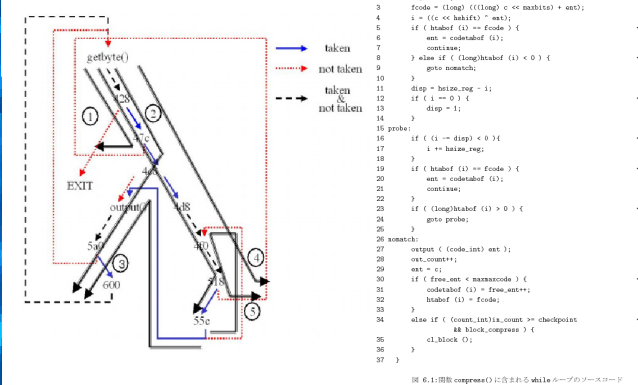
## Extracted Information

- Trace IDs are displayed in the order of their frequency
- Train data set case
- Traces in the function compress

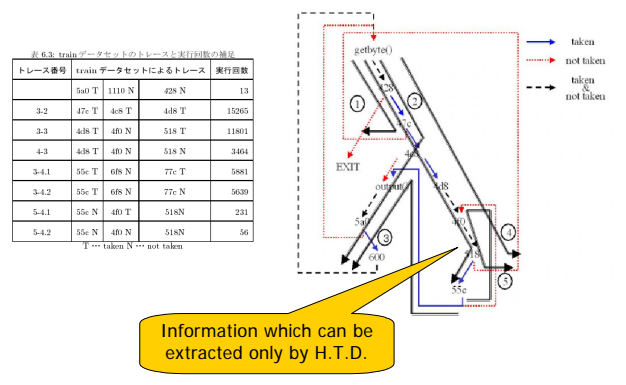
表 6.2: train データセットによるトレースと実行回数

トレース番号	train データセットによるトレース	実行回数
1	1110 N 428 T 47c N 138063	
	428 T 47c N 1110 N 138551	
	47c N 1110 N 428 T 138551	
	1210 T 1210 T 1210 T 131735	
	81c T 81c T 81c T 117750	
2-1	1110 N 428 T 47c T 114692	
	c00 N c00 N c00 N 110917	
	5a0 N 1110 N 428 T 110904	
	848 N 858 N 5a0 N 110817	
	858 N 5a0 N 1110 N 110804	
	al4 T a08 T 10ac T 110723	
	10ac T 10ac T 10ac T 107725	
	428 T 47c T 408 N 99397	
2-2	47c T 408 N 608 N 99397	
	7e4 T 848 N 858 N 97087	

T ... taken N ... not taken



12



13

## Results

- Hot Trace was extracted from the program compress successfully
- These information can be used in the compiler optimization such as *speculative instruction scheduling* and *speculative thread initiation*
- Limited Trace ID length limits the performance

14

## Conclusions

- We have developed *Hot Trace Detector* as a support architecture for *Advanced Parallelizing Compiler*
- FPGA based experimental machine REX allows us an efficient experiment
- We have demonstrated that frequent traces can be extracted from programs by using Hot Trace Detector

15