



## Employing on-chip comparator circuits for detecting errors

Dr. Oğuz Ergin,  
TOBB University of Economics and Technology

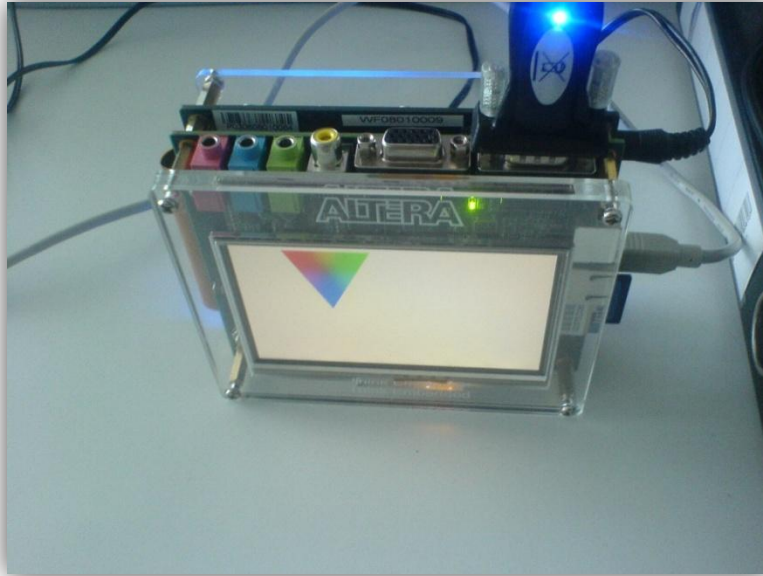
# Oğuz Ergin



- ▶ Phd from SUNY - Binghamton
- ▶ 15 months in Intel Barcelona Research Center
- ▶ Assistant Professor in TOBB University of Economics and Technology, Ankara, Turkey since January 2006
- ▶ Directing the Kasirga research group.



# Embedded Hardware Design



## ▶ Chip Design and Production

- ▶ Graphics processor
  - OpenGL ES 1.1
  - 50 Mhz (on FPGA)
- ▶ IP Core Design
  - ▶ AES Encryption
  - ▶ Image Processing
  - ▶ RS232 Serial Port Controller
  - ▶ VGA controller
  - ▶ Keyboard controller
- ▶ Kasirga Processors



# Supporting Software



## ► Kasirga İşlemcisi Derleyicisi - Tulpar (.NET – C#)

The screenshot displays the Tulpar 4.0 IDE interface. The main window is divided into several panes:

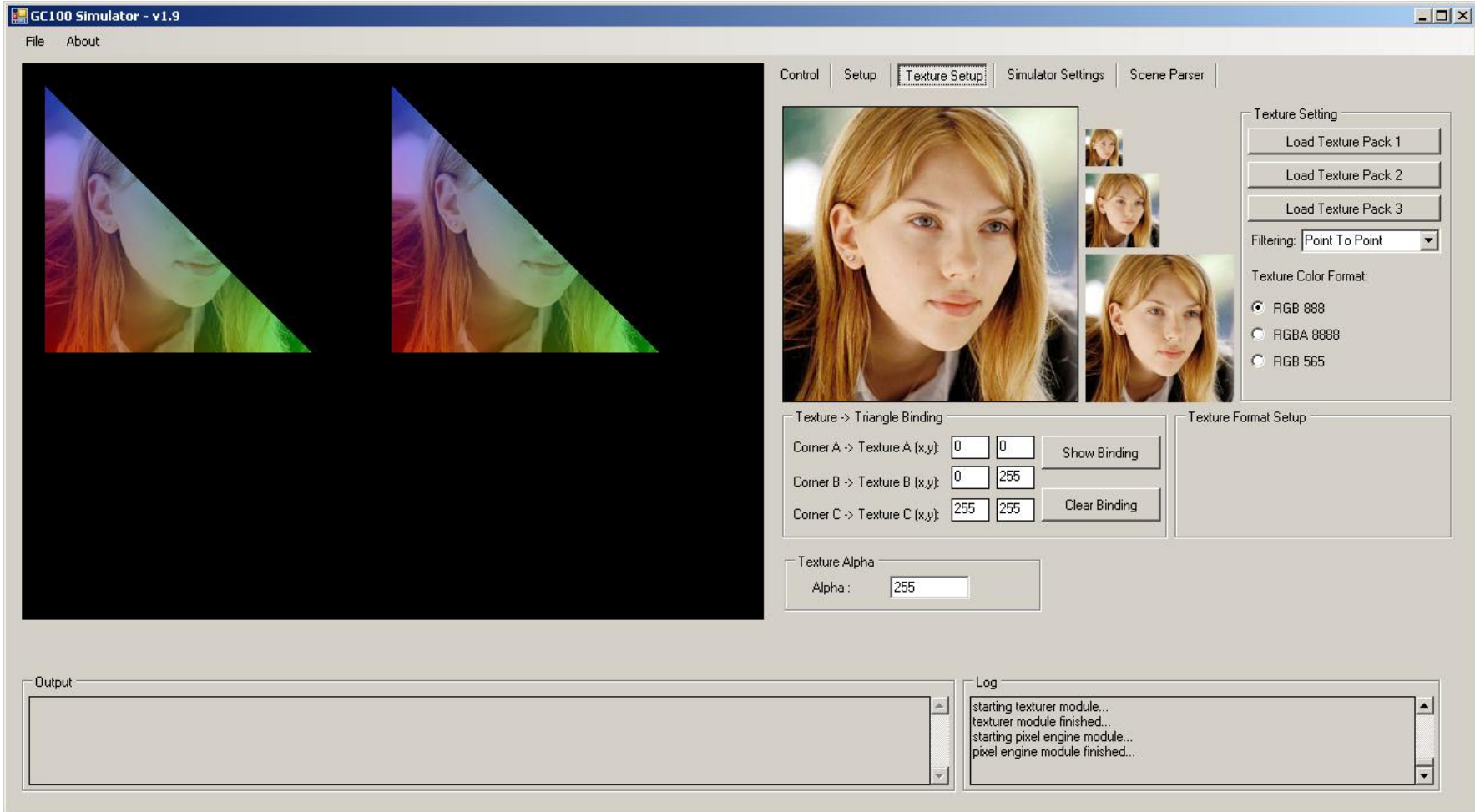
- Source Code (Left):** C# code for a program that calculates the sum of digits of a number. The code includes a `while` loop for digit extraction and a `switch` statement for digit classification.
- Assembly Code (Middle):** The corresponding assembly code generated by the compiler, showing instructions like `mov r3,r0`, `cmp r3,r4`, and `add r5,r5,r0`.
- Debugger (Right):** A window showing the current instruction being executed: `cmp r1 r2`. Below it, a register window displays the values of registers Y0 through Y7, and a stack window shows the return value `PC_Ret`.

The interface also includes a menu bar with options like `Dil` (Language) and `Öykünme Şekli` (Debugging Style). The status bar at the bottom shows the current language is set to `English`.

# Supporting Software



## ► Benzetim yazılımları (Java, .NET – C#)





# Computer Architecture

---

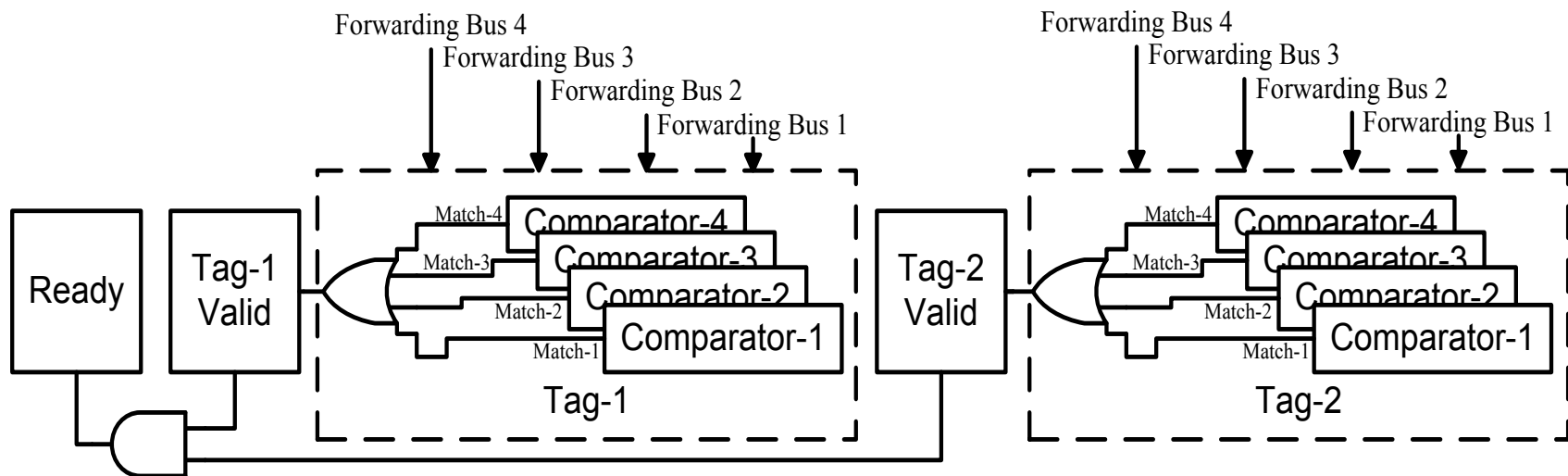
- ▶ **Reliability**
  - ▶ Soft Errors
  - ▶ Hard errors
  - ▶ Variability
- ▶ **Energy Efficiency**
- ▶ **High Performance**
- ▶ **Complexity-effective design**





# Comparator Circuits

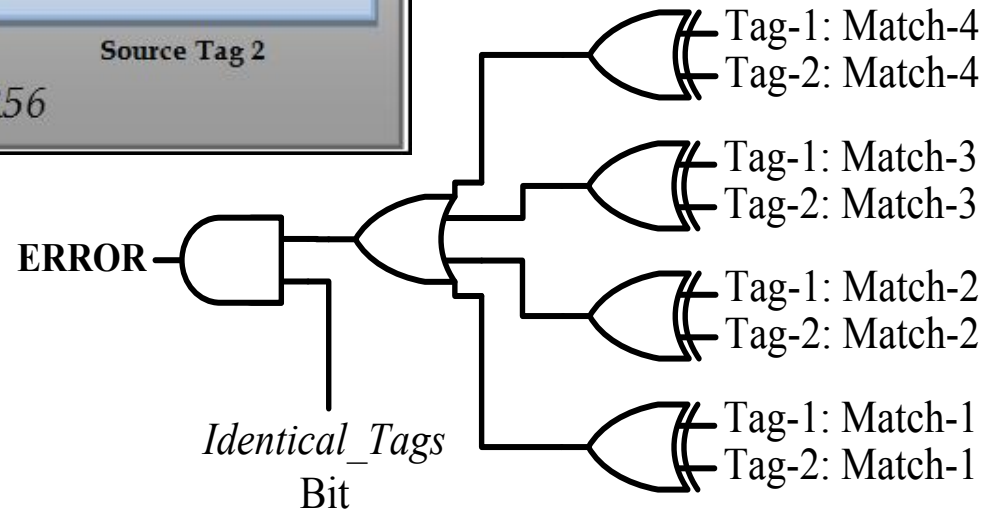
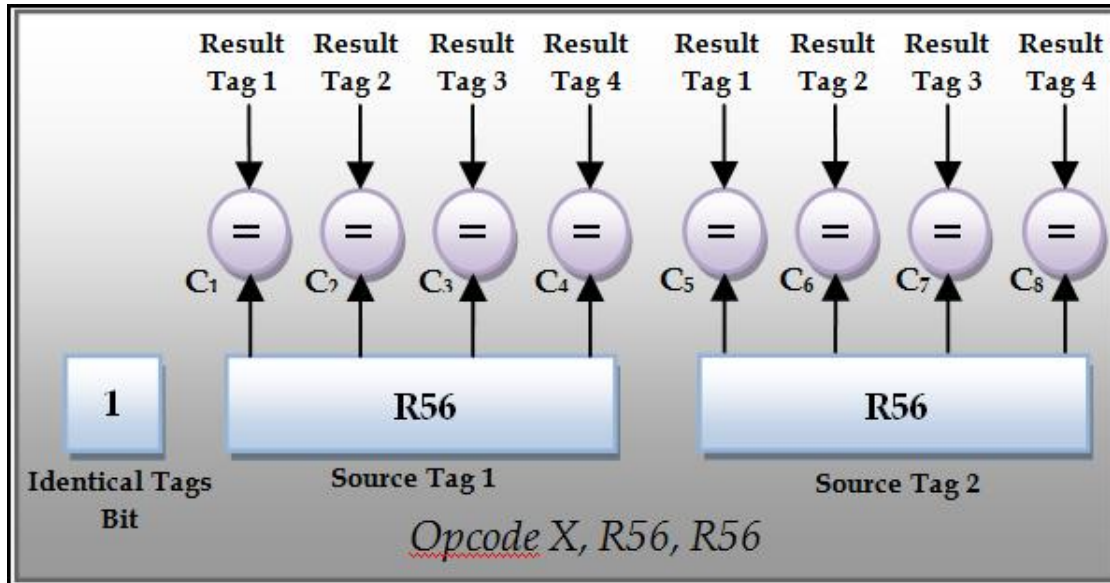
- ▶ Used inside the issue queue to keep track of dependencies
- ▶ 1 comparator for each forwarding bus per tag





# Exploiting Comparators

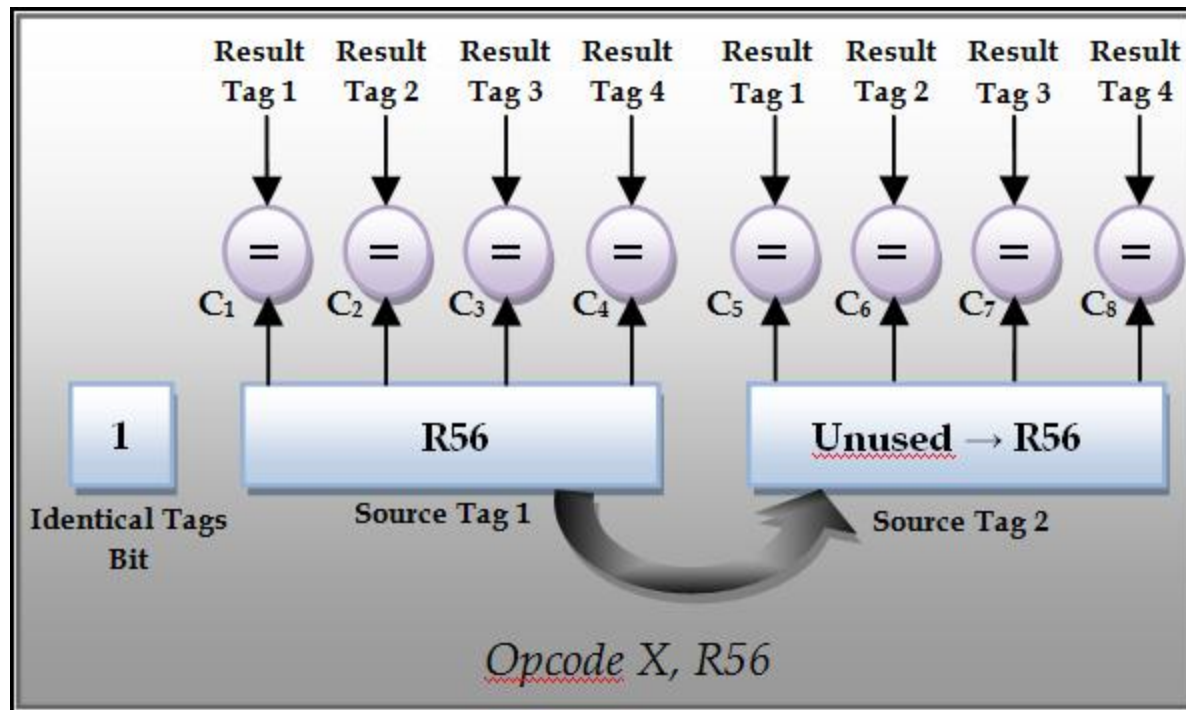
## ▶ Instructions with identical tags





# Exploiting Comparators

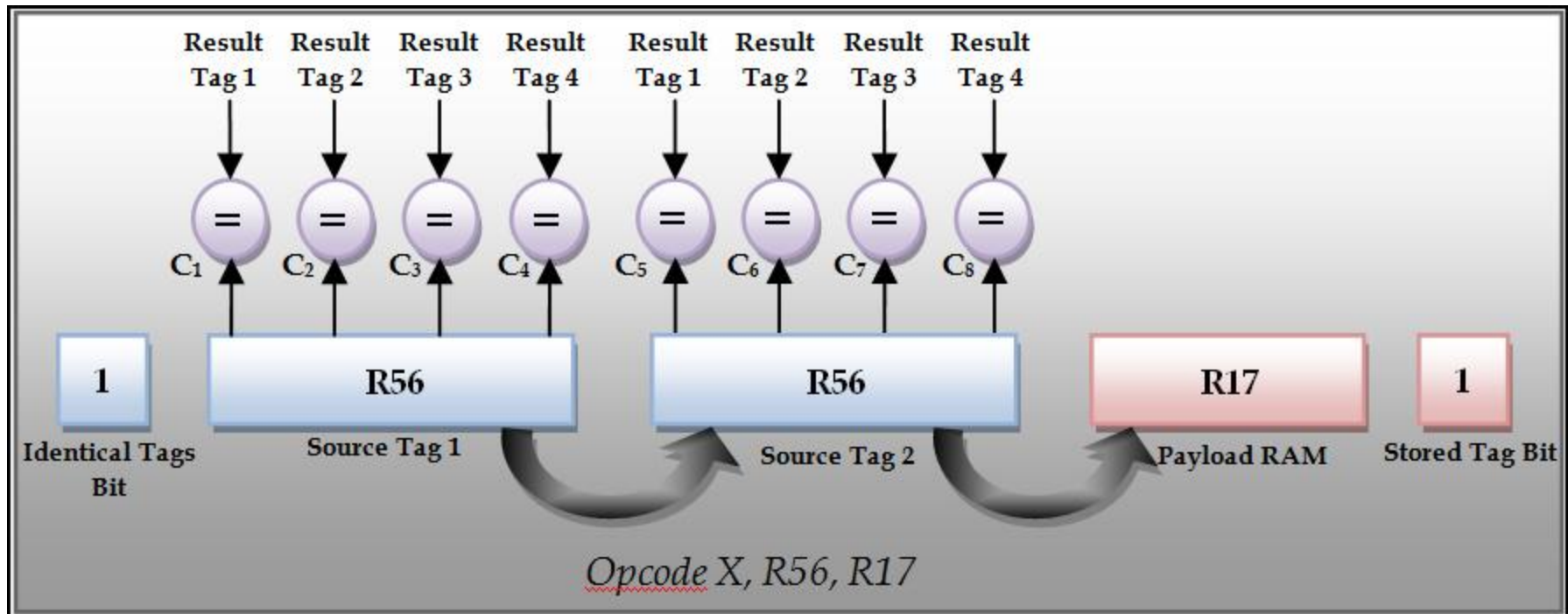
## ► Instructions with single operand





# Exploiting Comparators

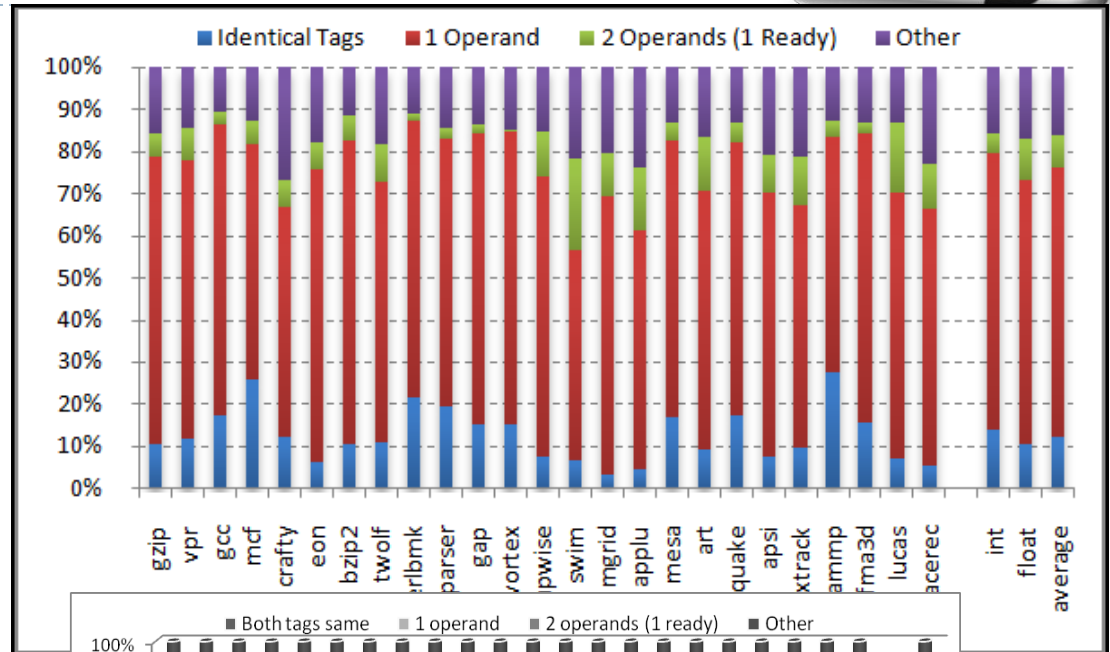
- ▶ Instructions with one ready source



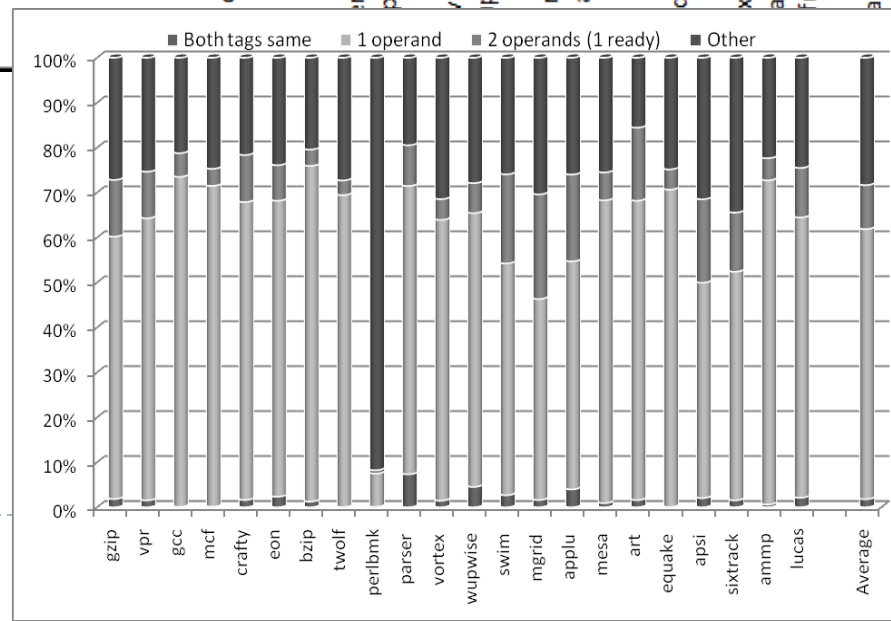
# Statistics



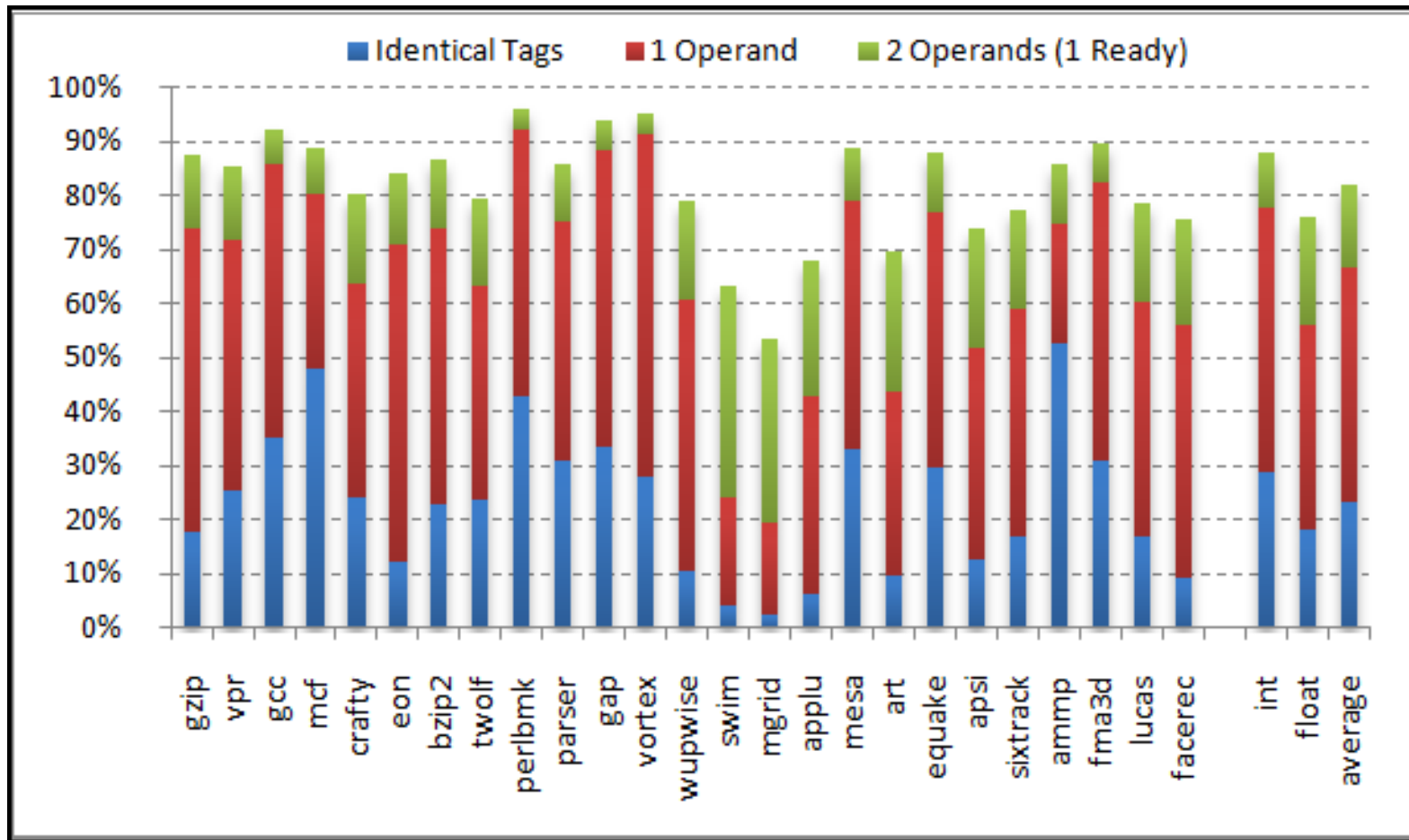
## ▶ PTLsim (x86)



## ▶ Msim (Alpha)



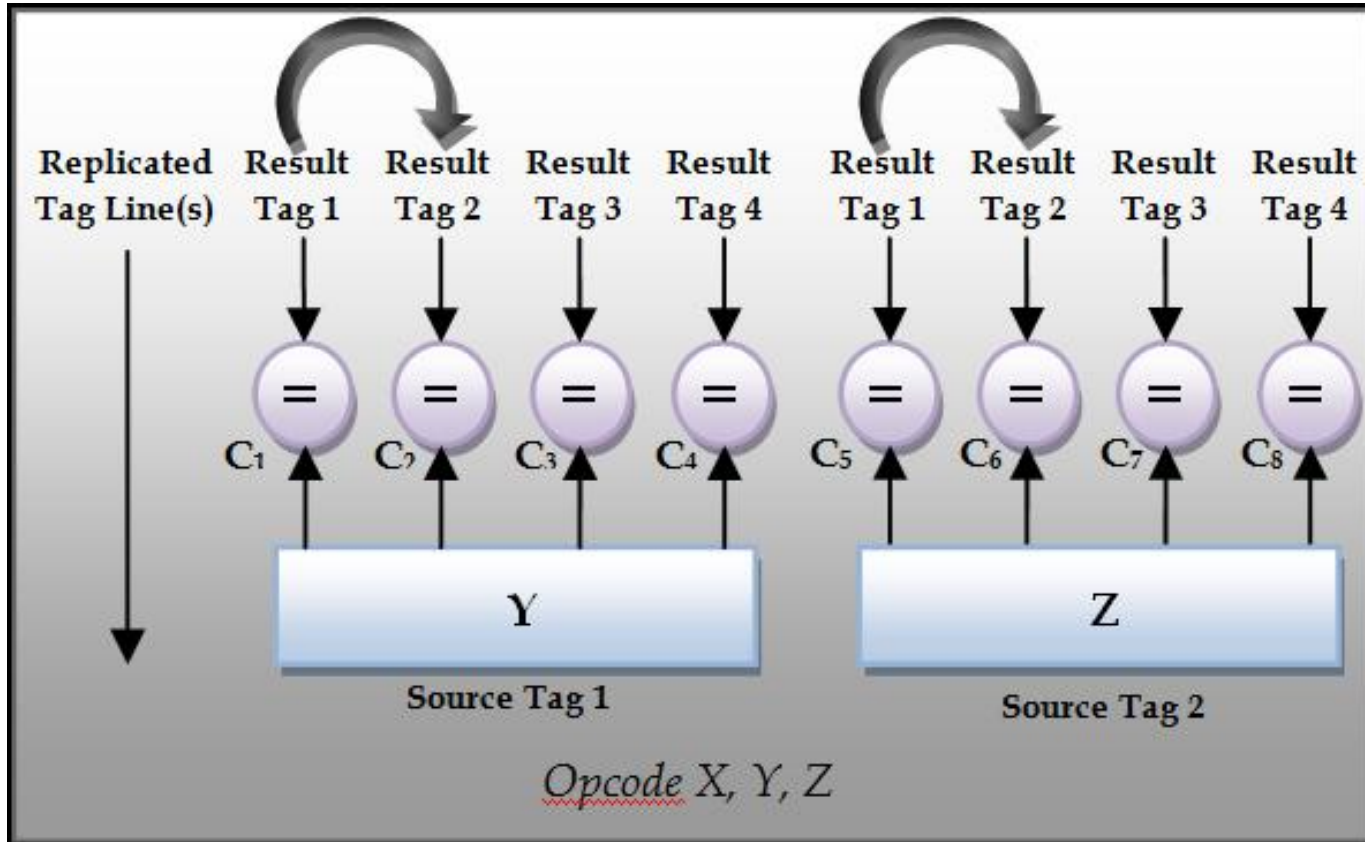
# Detection Coverage



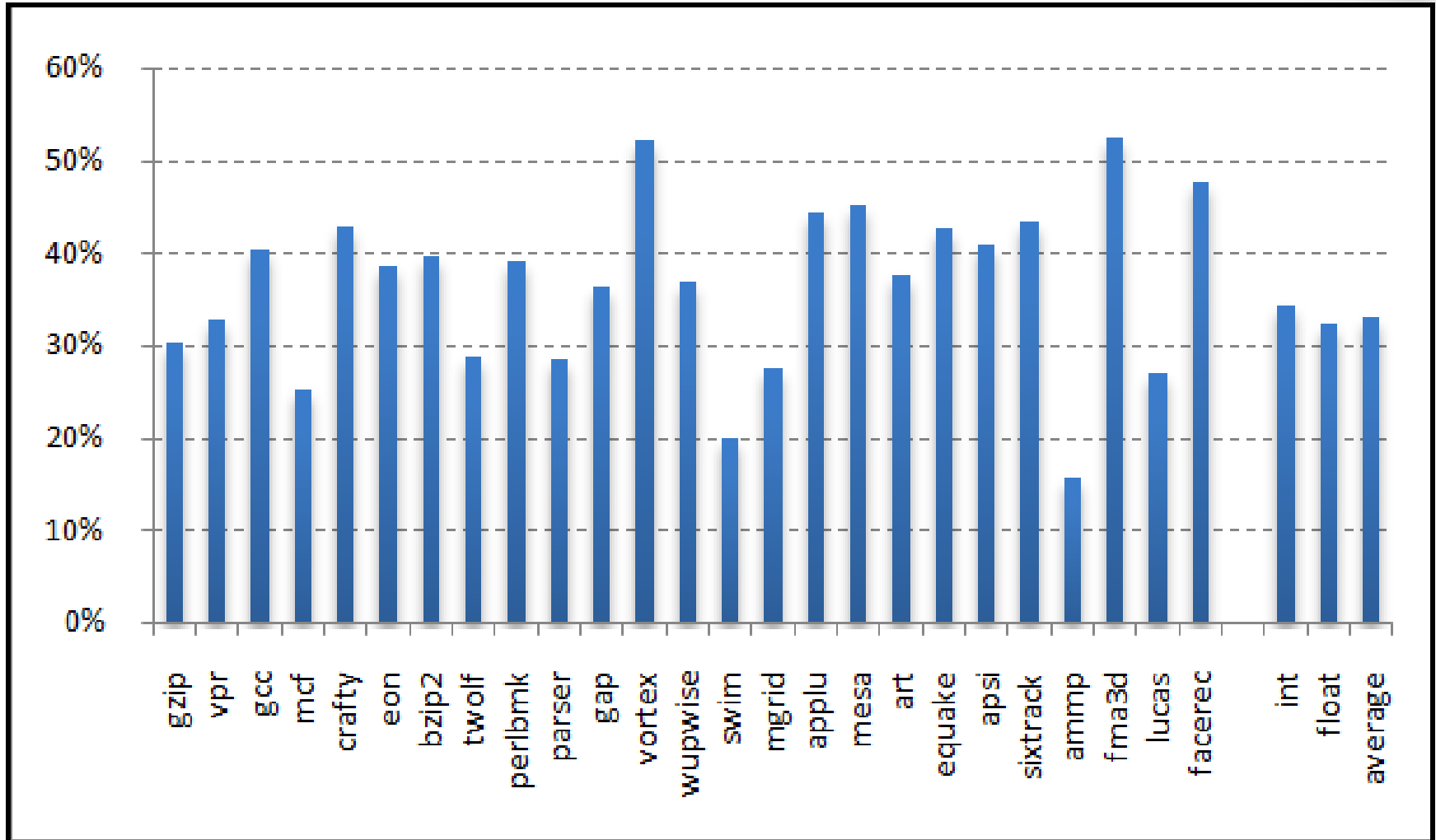


# Exploiting Comparators

## ▶ Unemployed Forwarding Logic



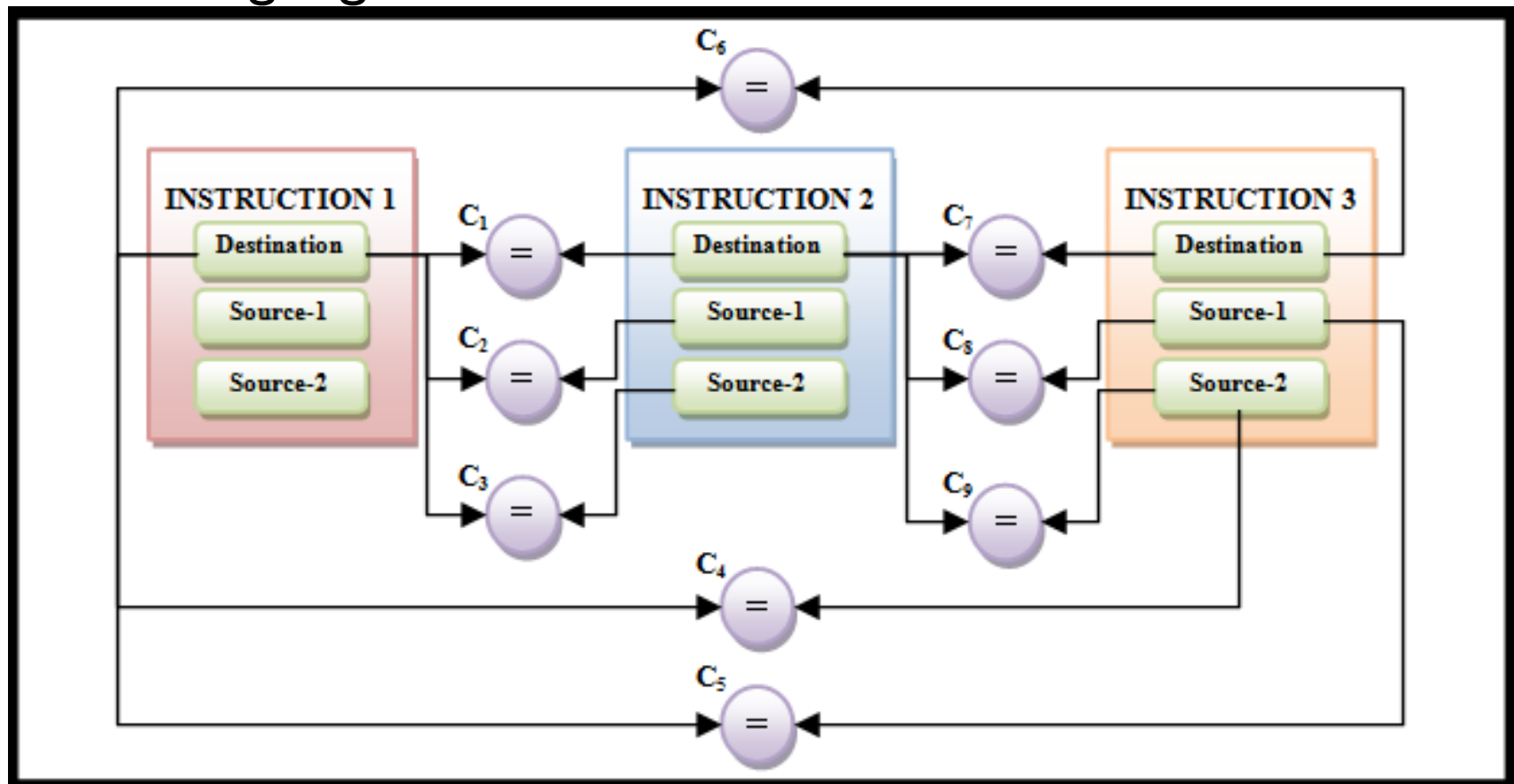
# Soft Error Coverage Results





# Comparator Circuits

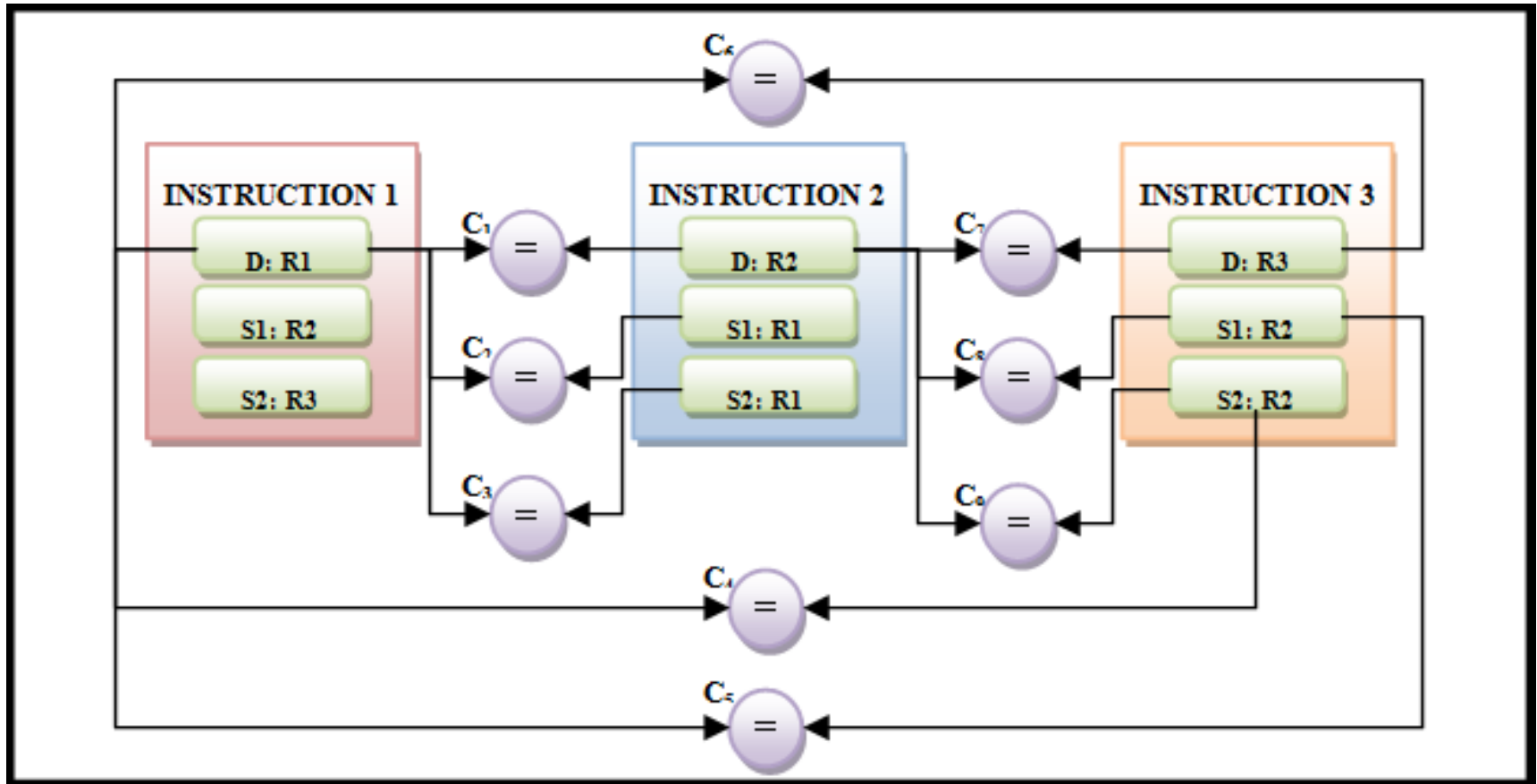
- ▶ Also used at the rename stage inside the dependency checking logic





# Exploiting Comparators

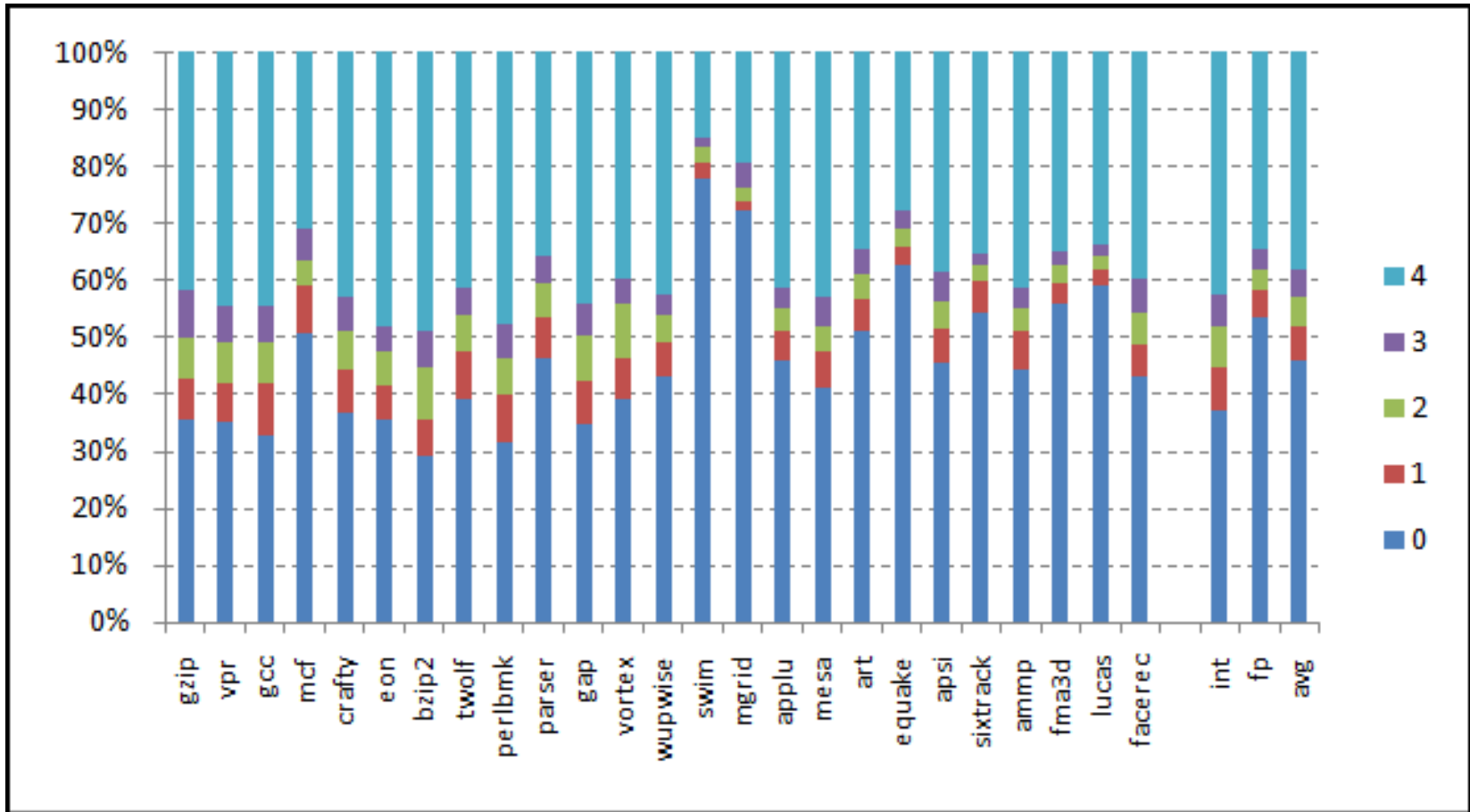
- ▶ When some of the rename slots are empty ...



# Statistics

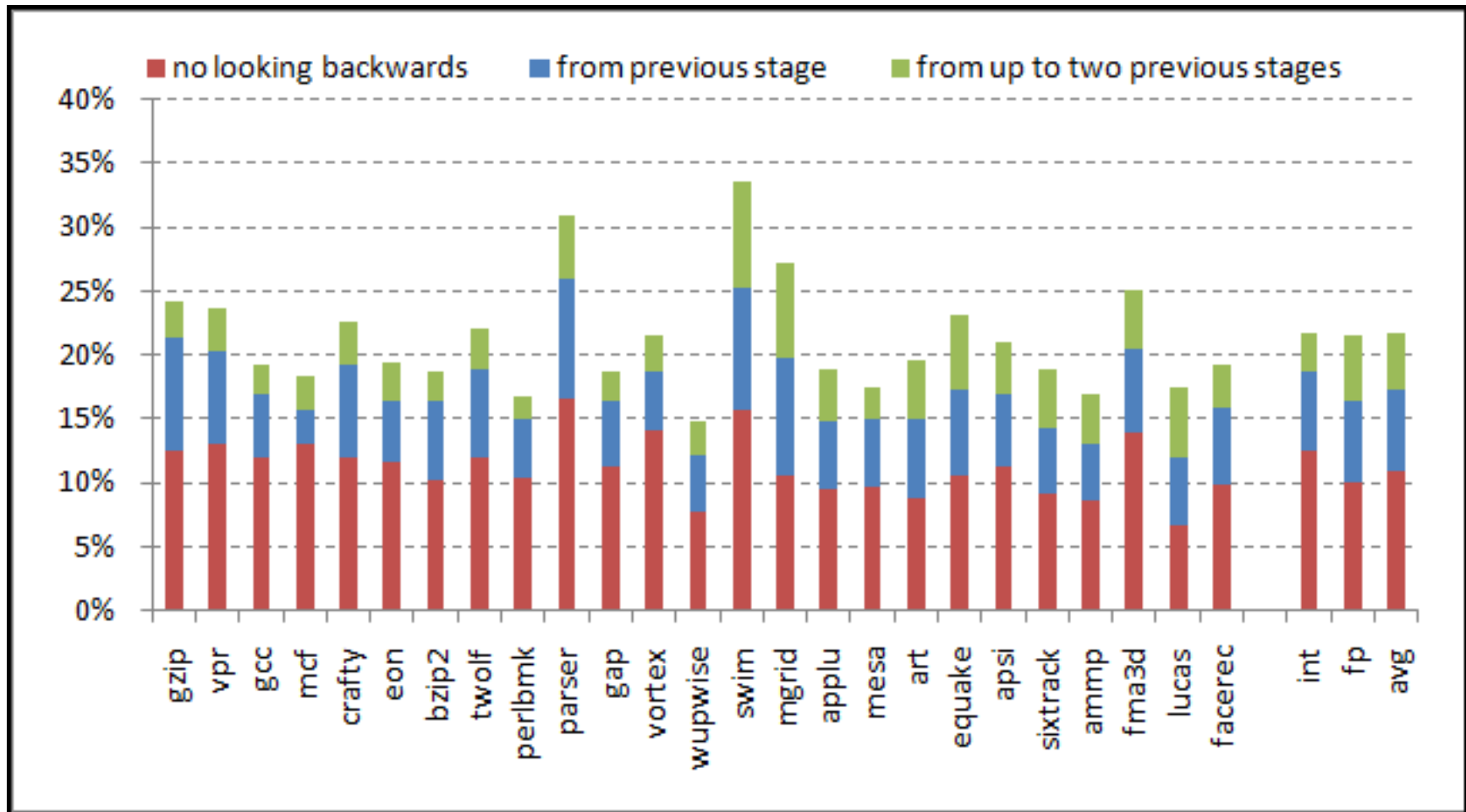


## ▶ Number of Rename Slots Used





# Soft Error Coverage





# Contact Information

---

- ▶ Kasirga Information Systems Ltd.
  - ▶ <http://www.kasirga.com.tr>
  - ▶ <http://kasirga.etu.edu.tr>
- ▶ Assistant Prof. Oğuz Ergin
  - ▶ [oergin@etu.edu.tr](mailto:oergin@etu.edu.tr)
  - ▶ [oguz.ergin@kasirga.com.tr](mailto:oguz.ergin@kasirga.com.tr)
  - ▶ Tel: +90 312 292 40 59
  - ▶ Cell: +90 532 384 61 94
  - ▶ Fax: +90 312 292 41 80

