

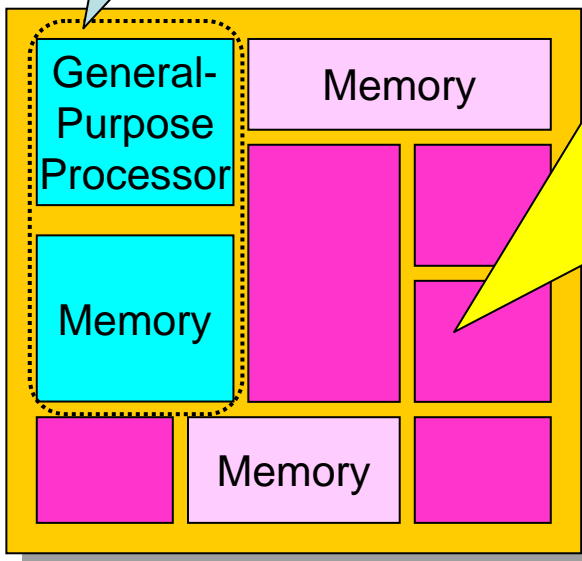
Redefis: An SoC Platform for Implementing Application- Specific or User-Custom Logic

Kazuaki Murakami
Kyushu University
redefis@fleets.jp

Five Ways to Design & Implement Custom Logic (from MPSoC'04)

How to Implement General Logic

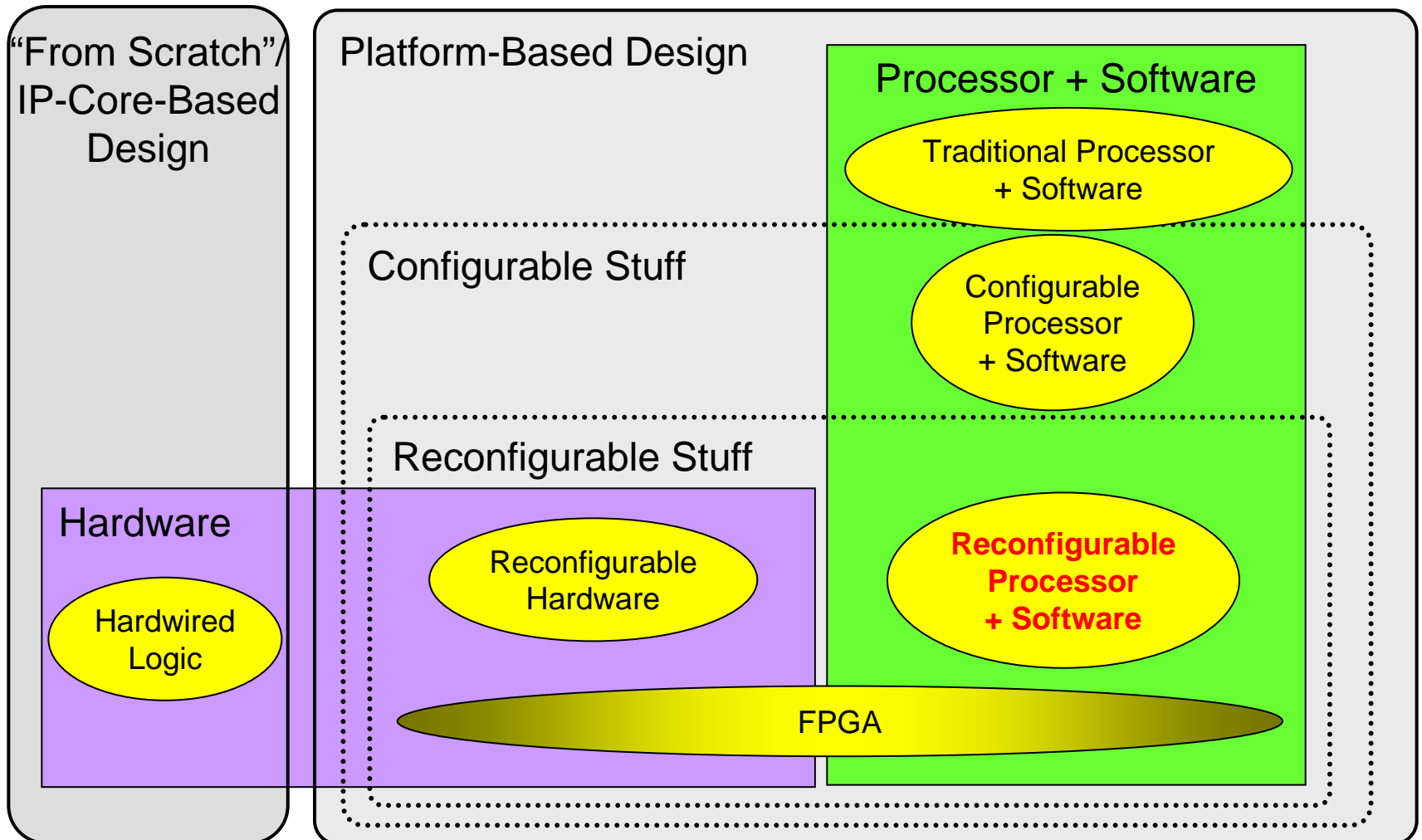
- General-Purpose Processor + Software



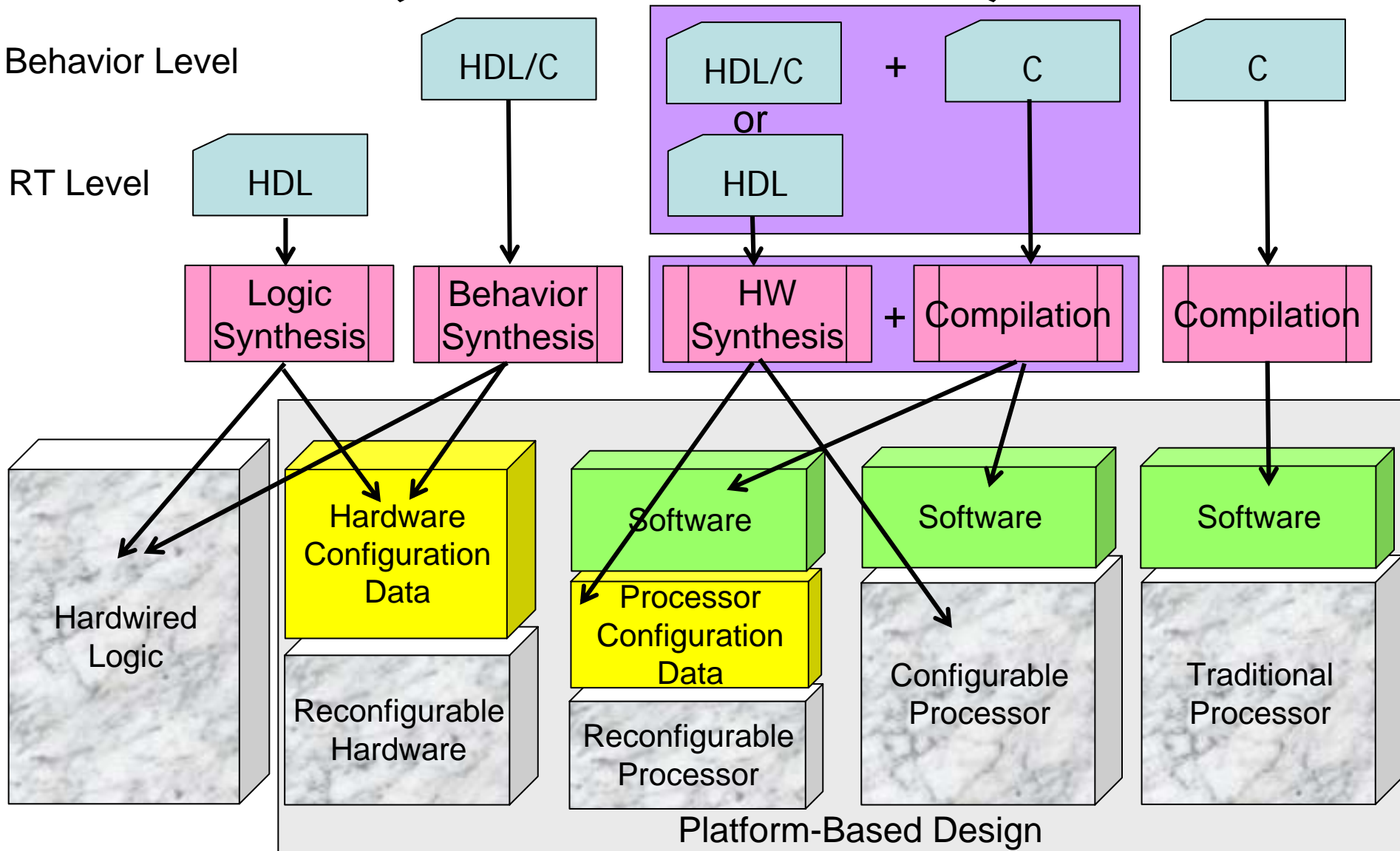
How to Implement Custom Logic

- **“From Scratch” Approach**
 - Design new hardware every time
- **IP-Core-Based Approach**
 - Reuse existing hardware designs, or IP cores
- **Platform-Based Approaches**
 - **Processor + Software**
 - **Configurable Processor + Software**
 - **Reconfigurable Hardware**
 - **Reconfigurable Processor + Software**

Five Ways to Design & Implement Custom Logic (from MPSoC'04)

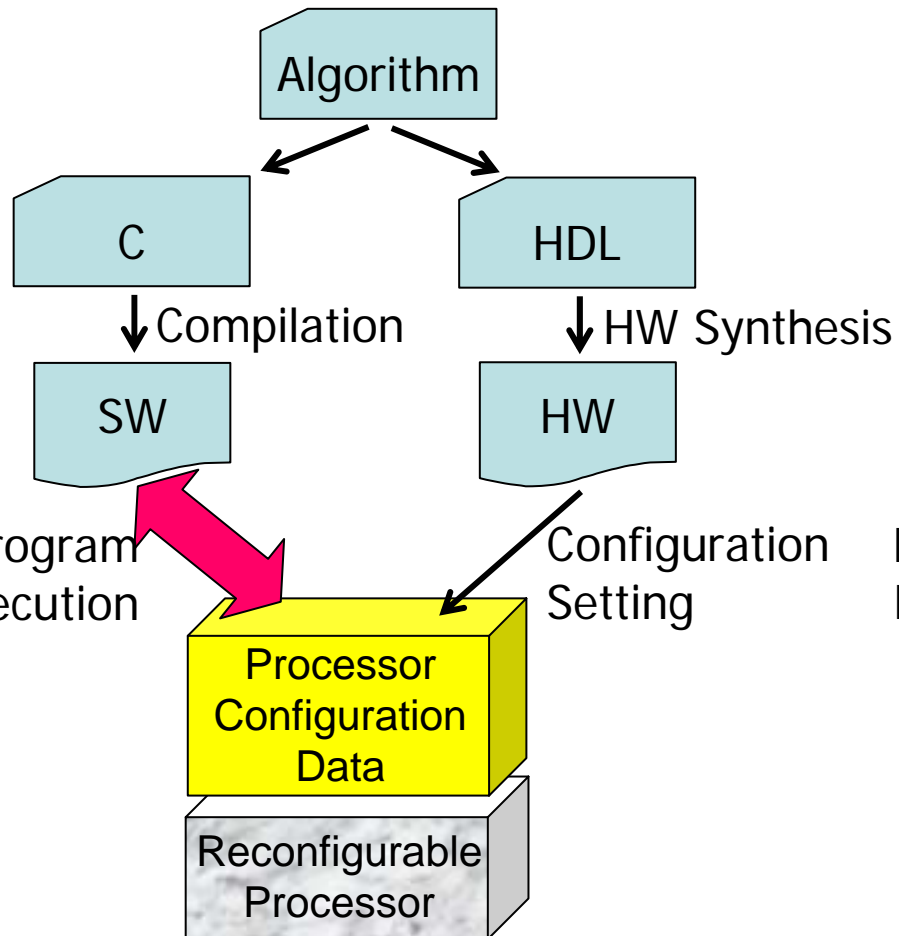


Five SoC Design Flows (from MPSoC'04)

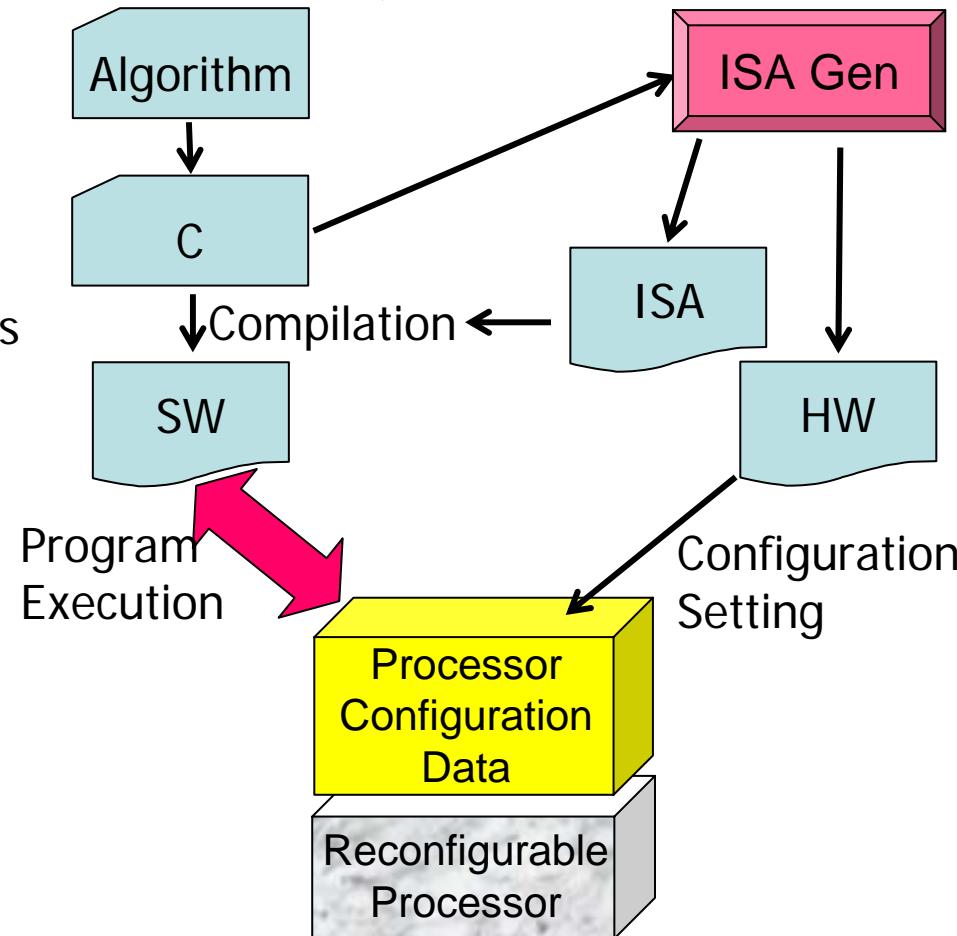


Redefis (Redefinable ISA Processor): A Reconfigurable Processor

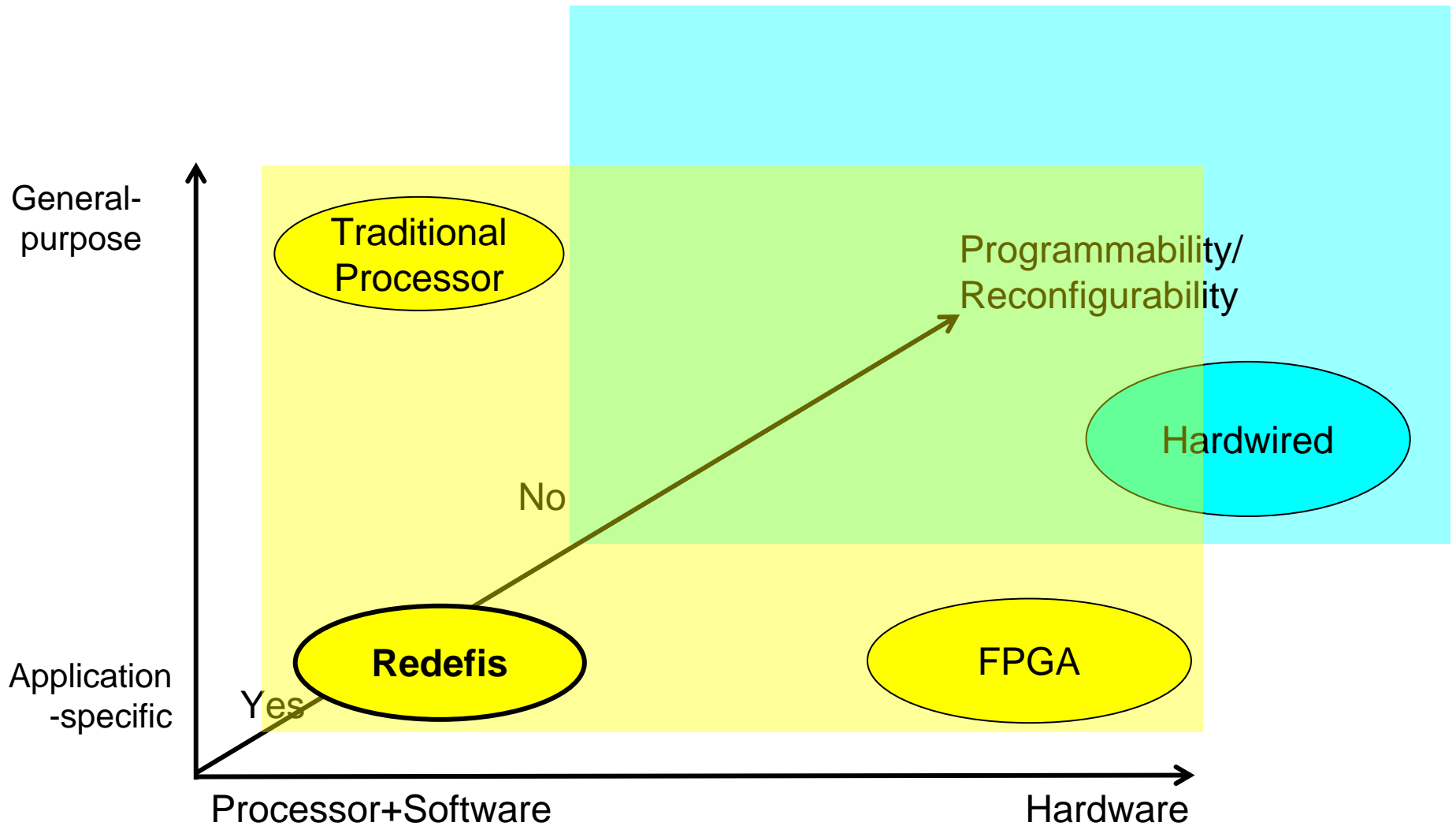
- Normal Reconfigurable Processor



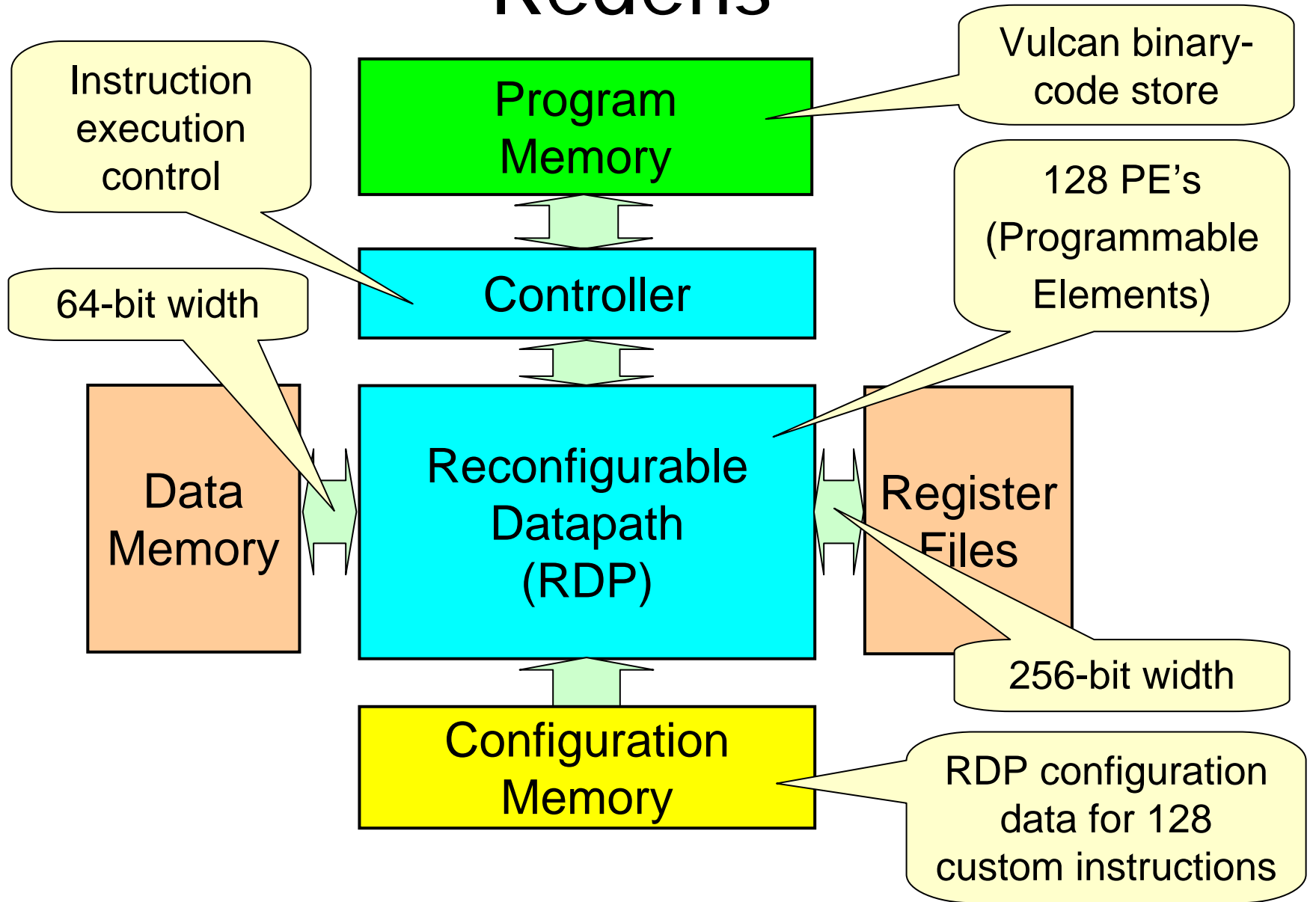
- Redefis (Redefinable ISA Processor)



Where Does Redefis Go?



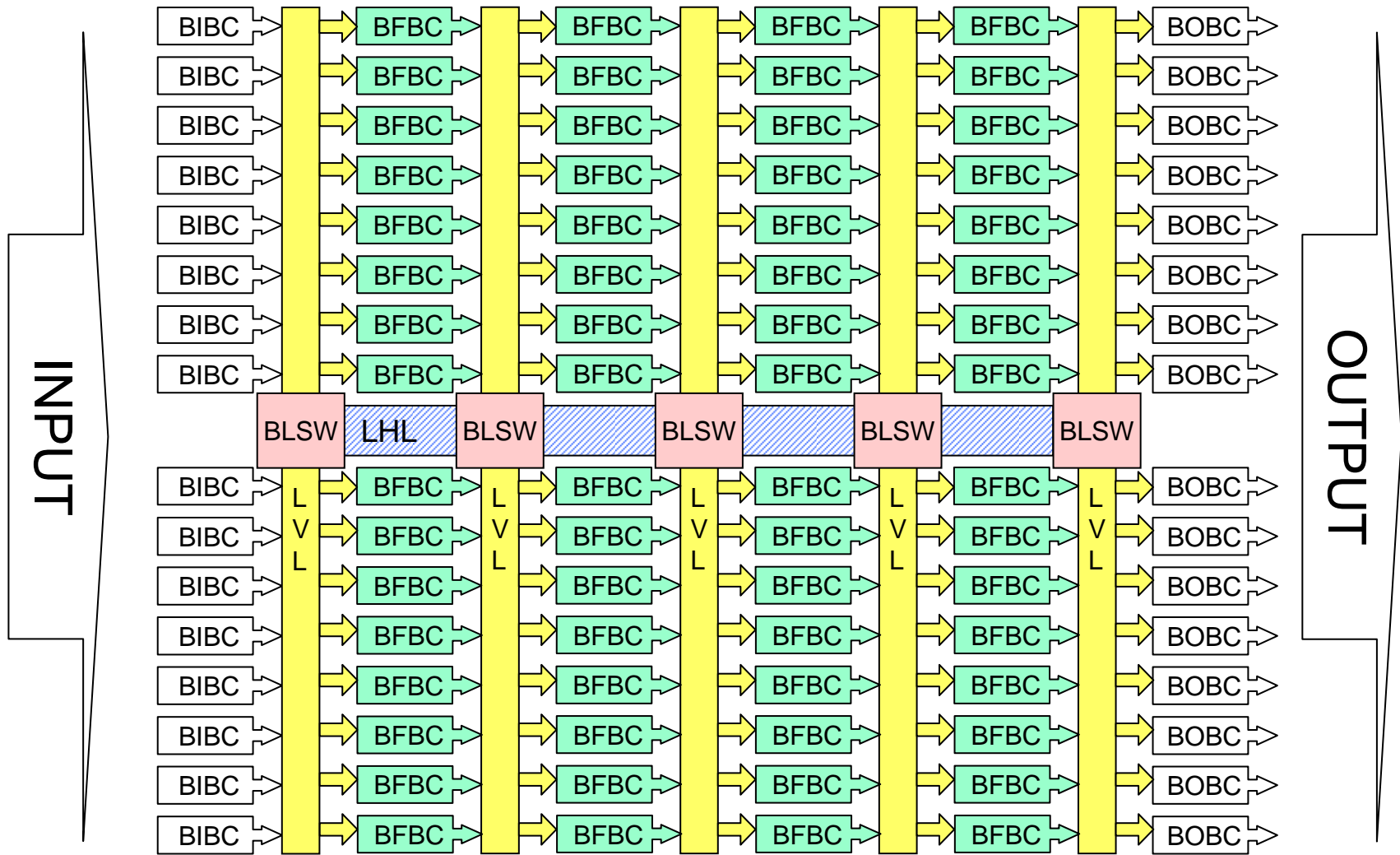
Vulcan: An Implementation of Redefis



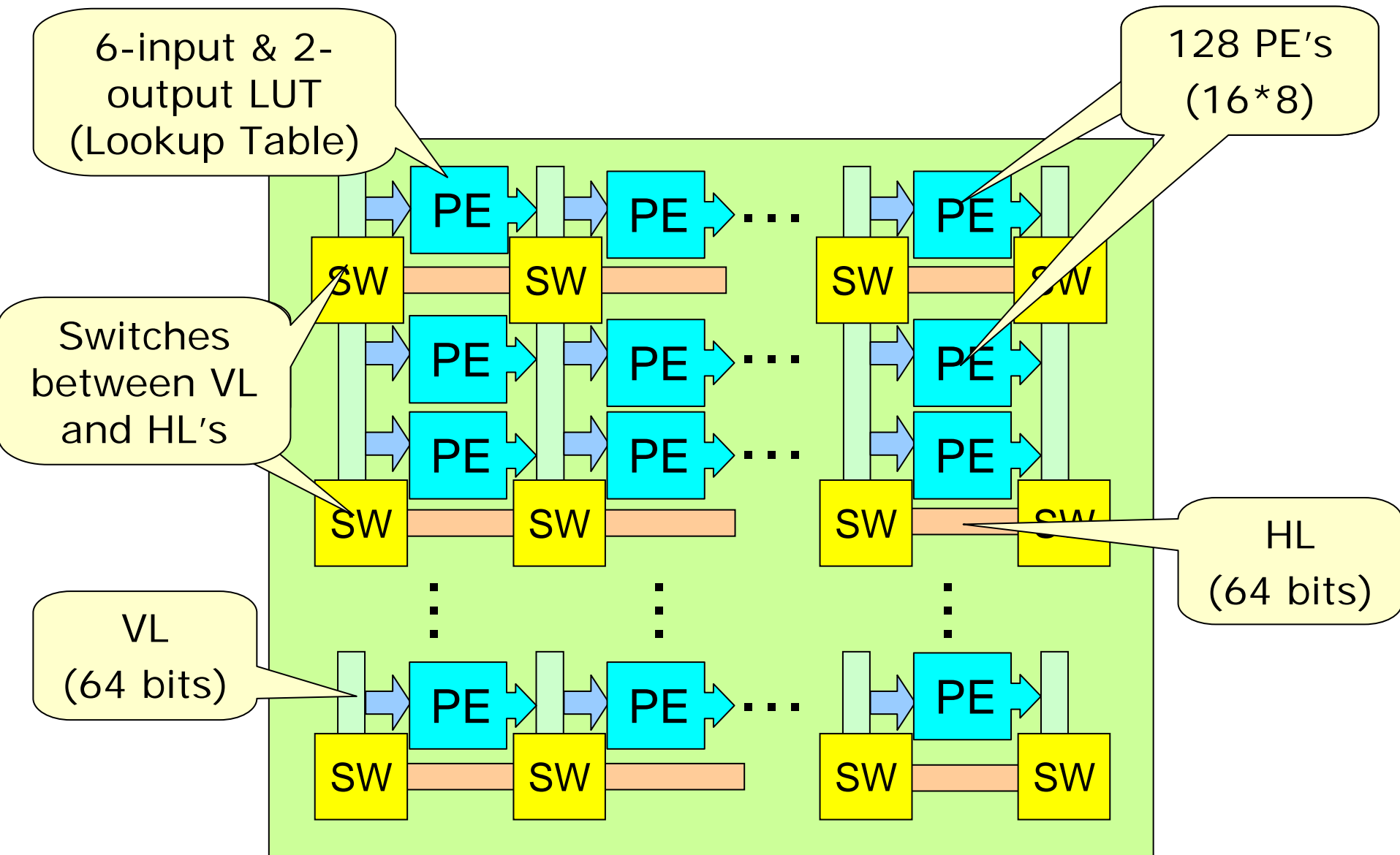
Vulcan Chip & Board



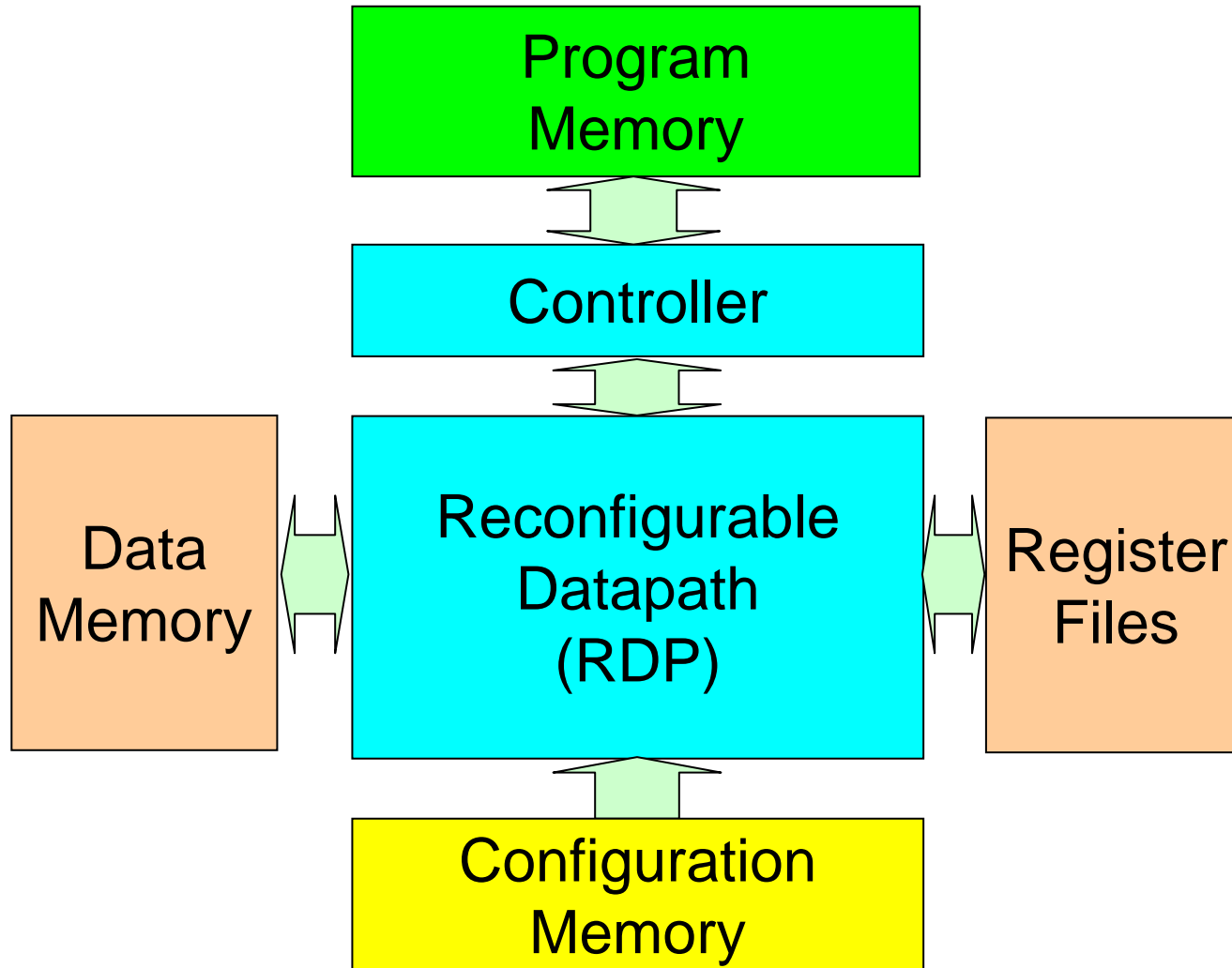
Reconfigurable Datapath (RDP)



Reconfigurable Datapath (RDP)

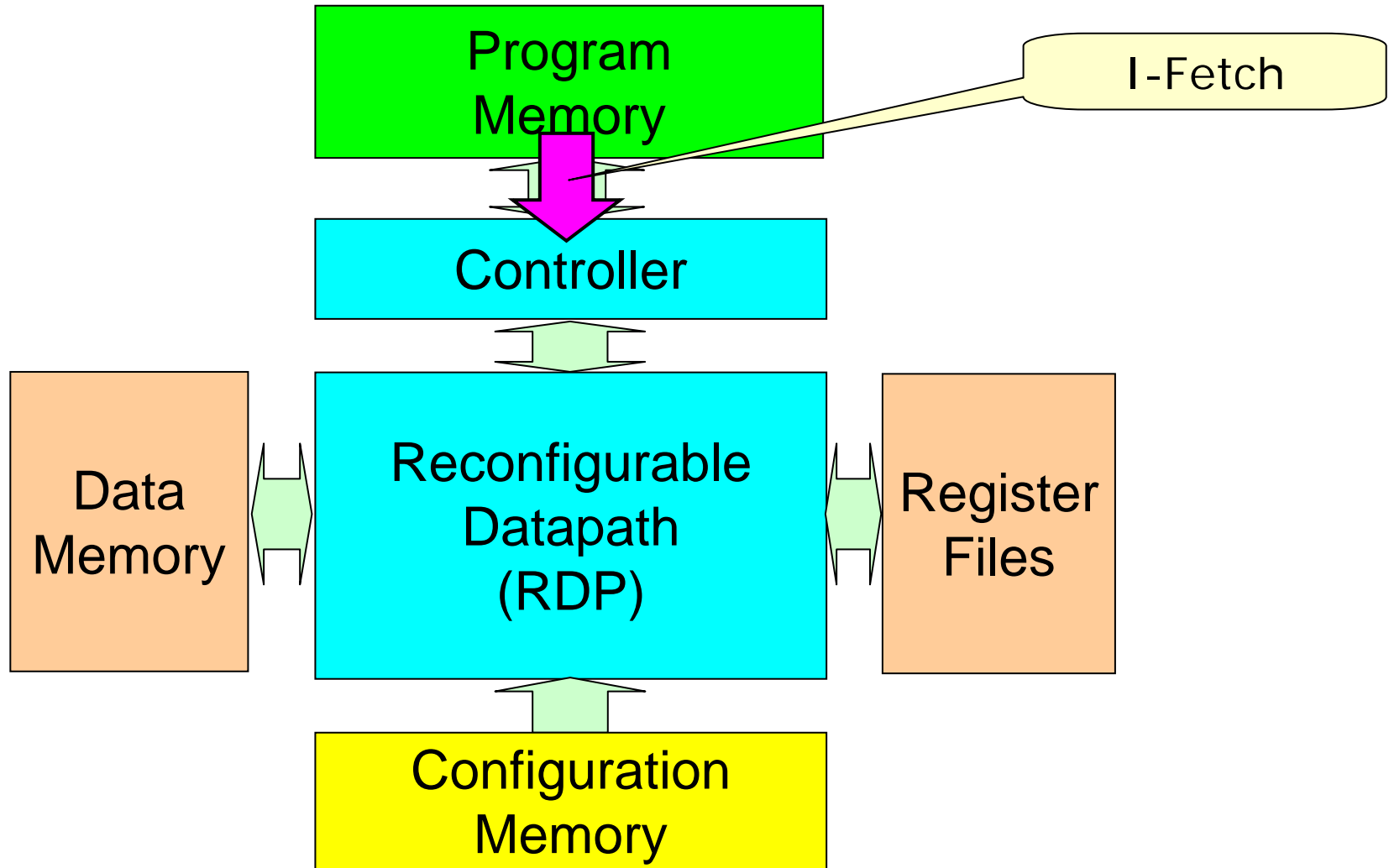


Instruction Execution Flow



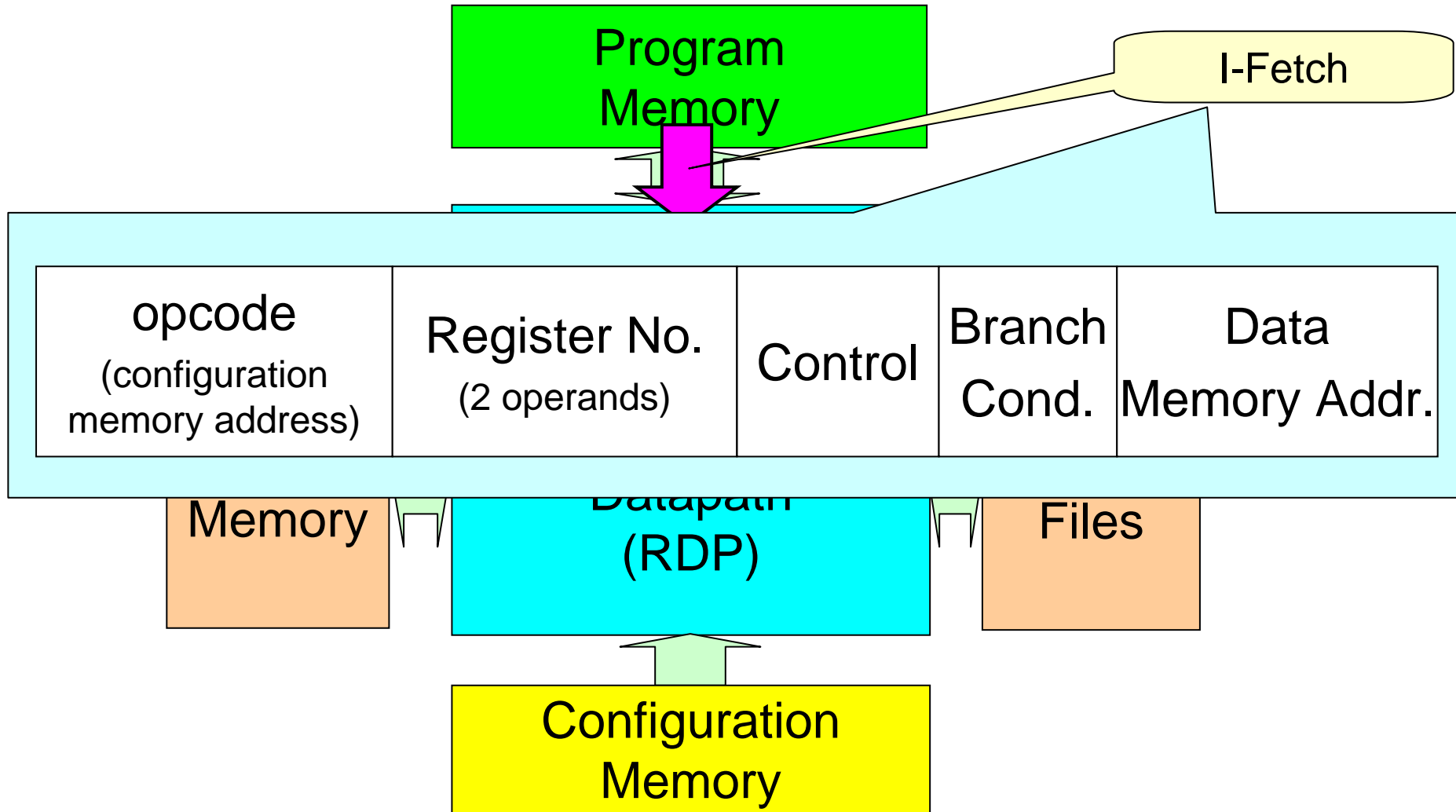
Instruction Execution Flow

- Phase 1: I-Fetch -



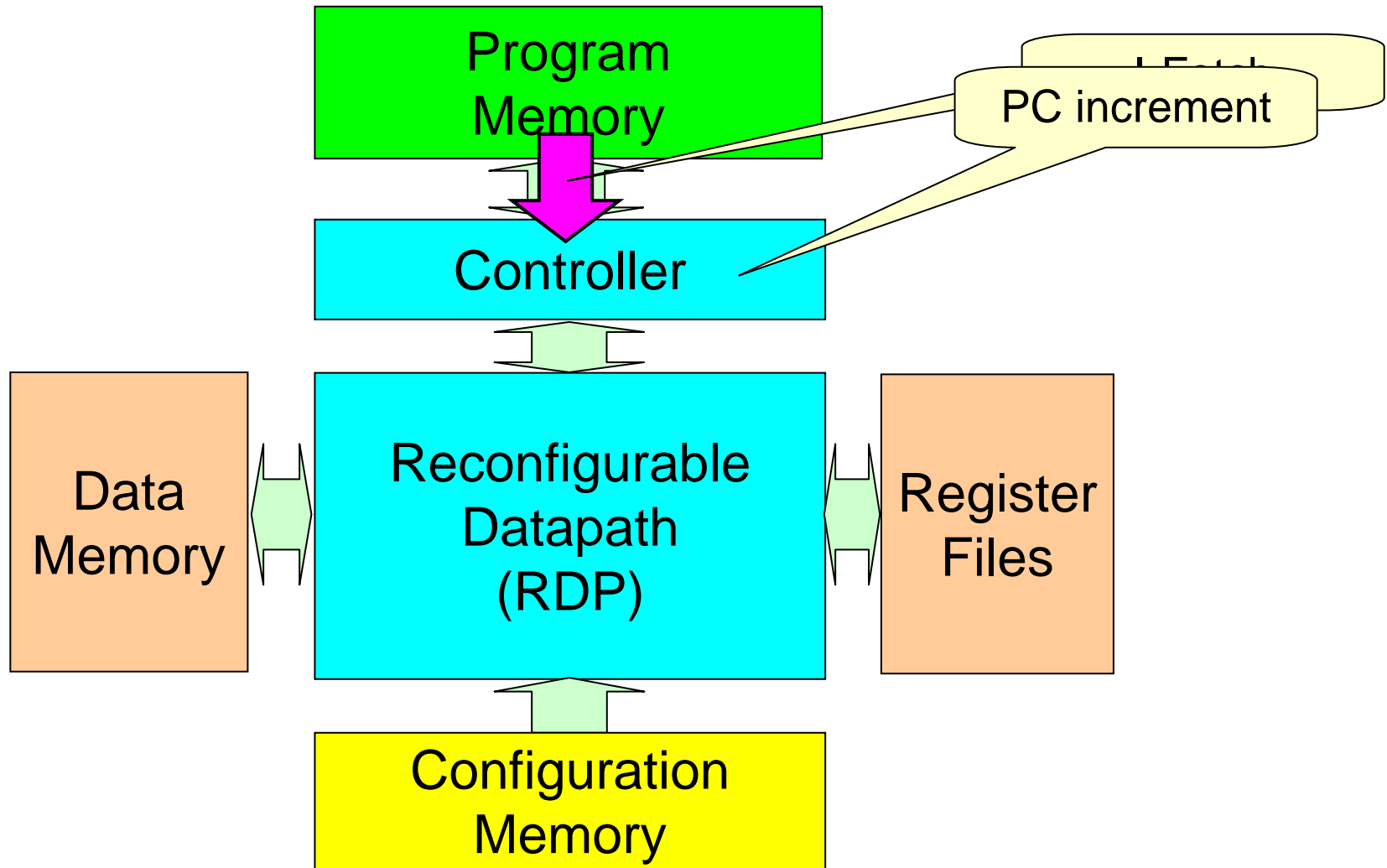
Instruction Execution Flow

- Phase 1: I-Fetch -



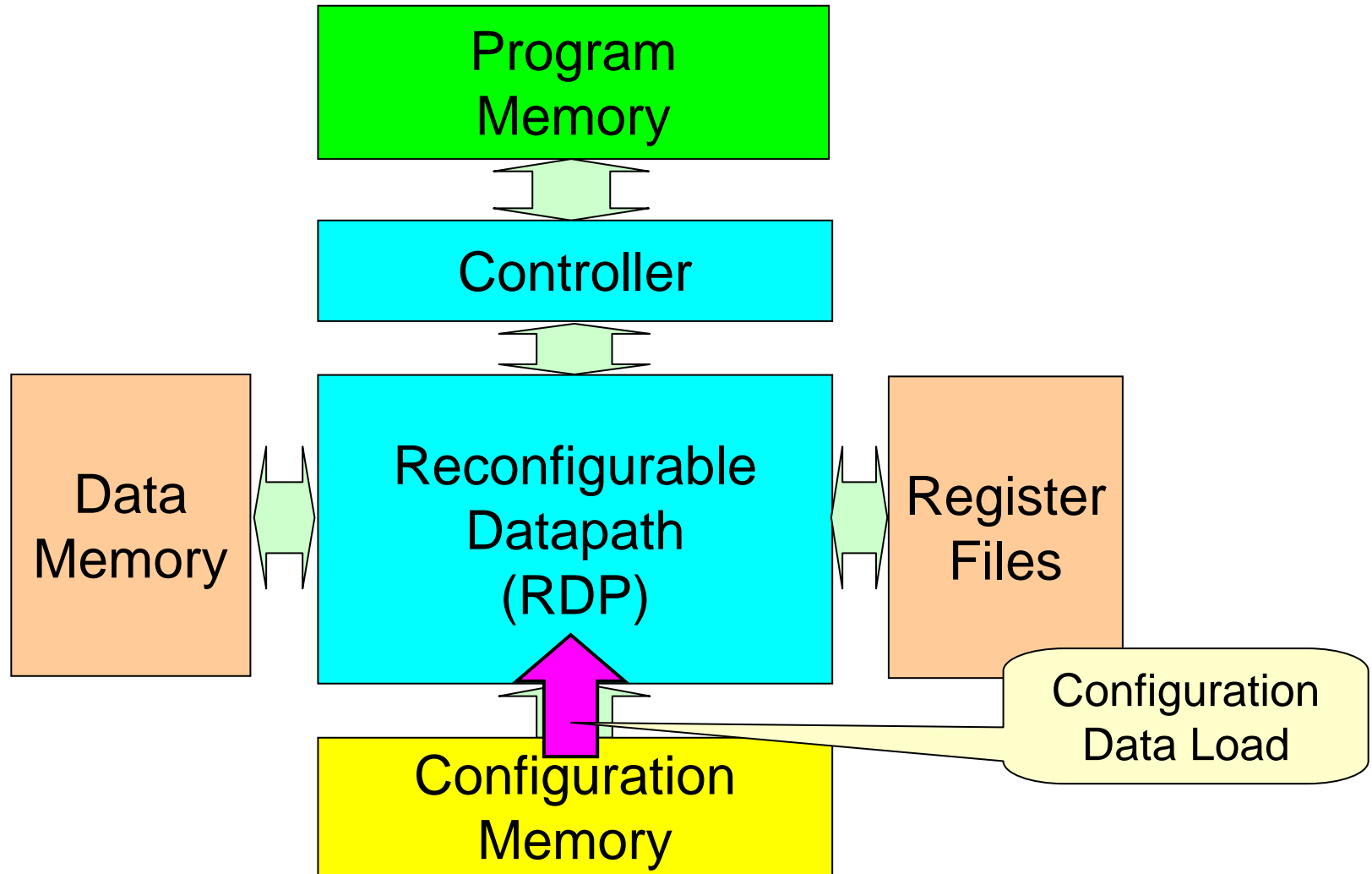
Instruction Execution Flow

- Phase 1: PC Update -



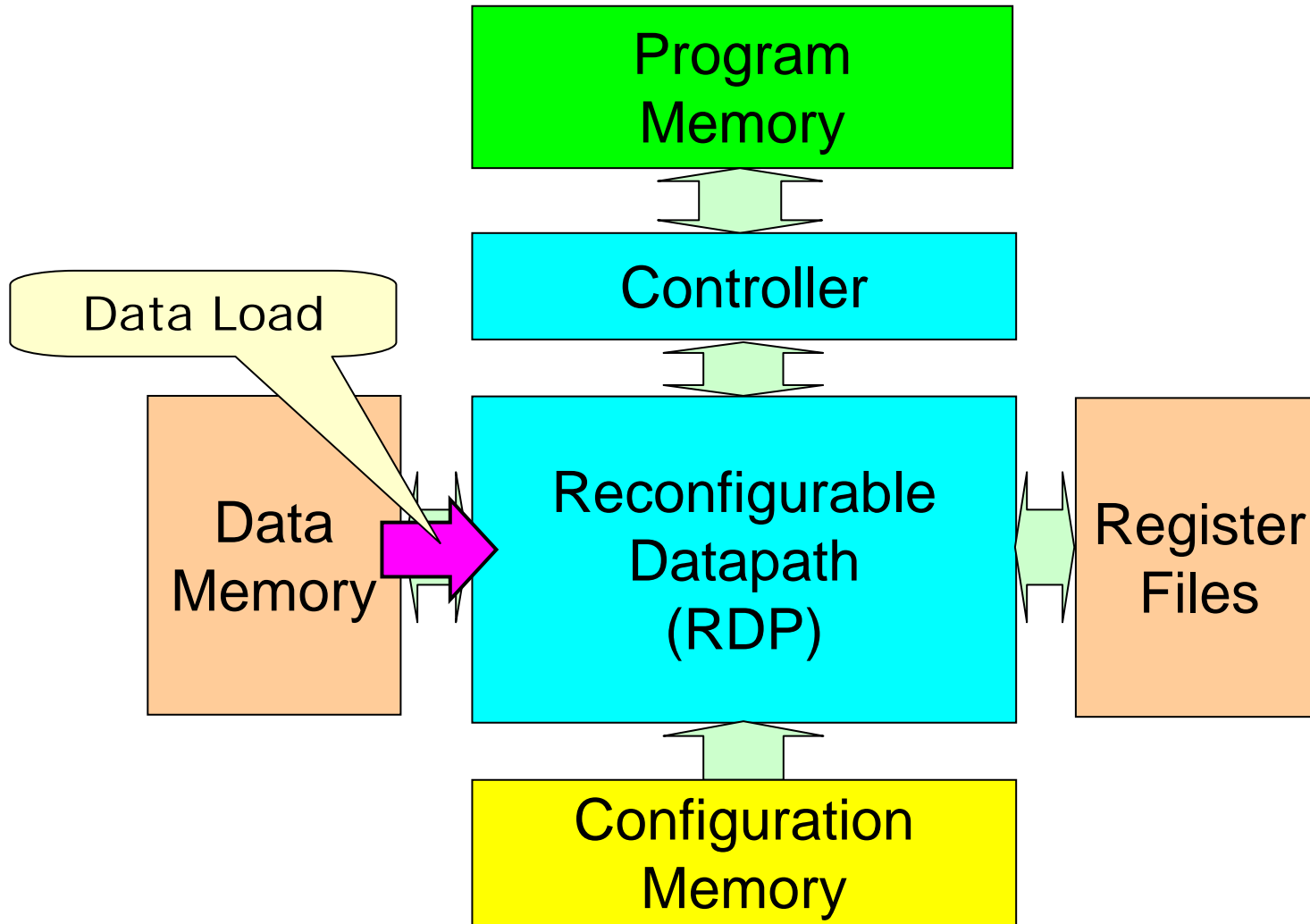
Instruction Execution Flow

- Phase 2: Configuration Data Load -



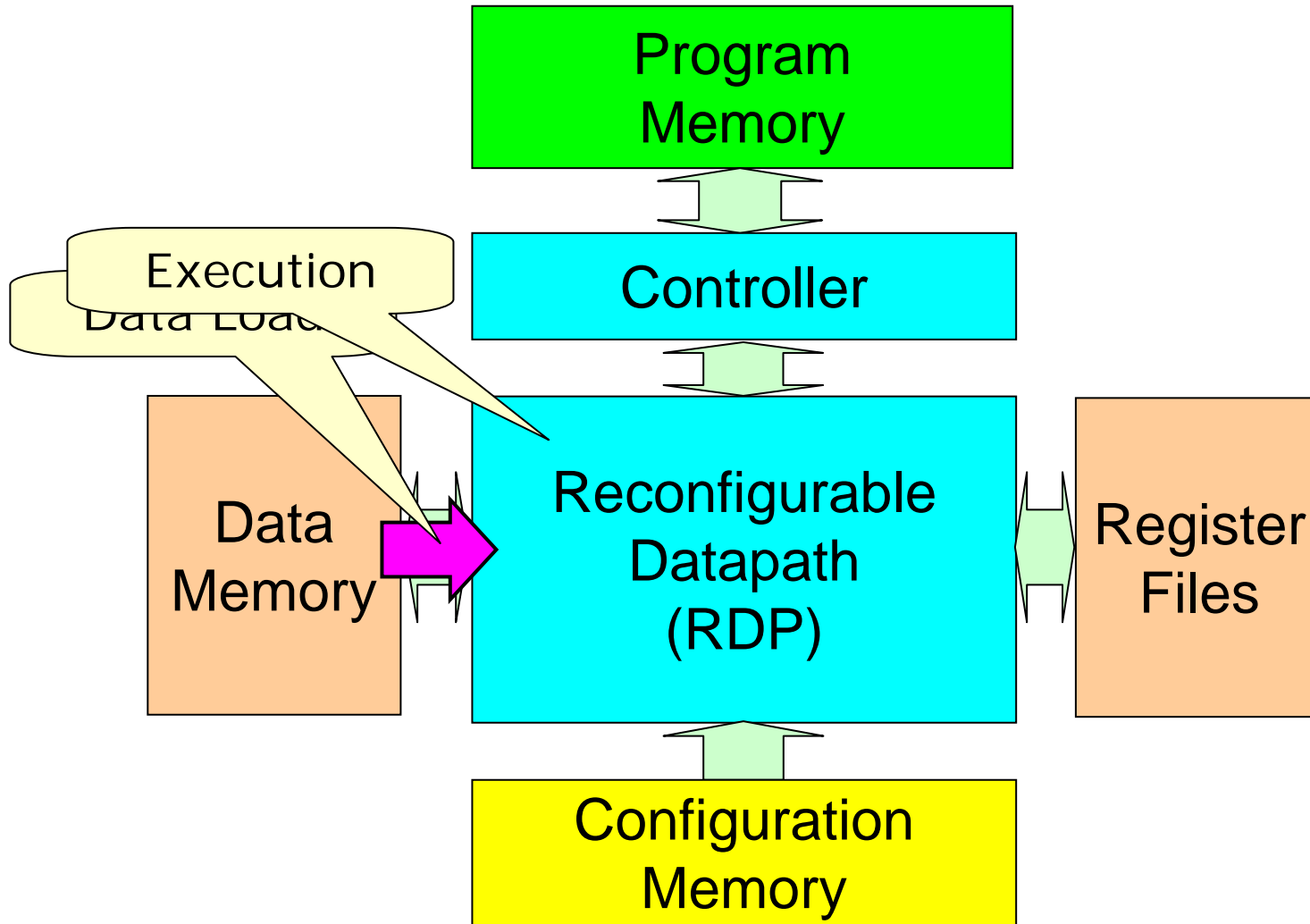
Instruction Execution Flow

- Phase 2: Data Load -



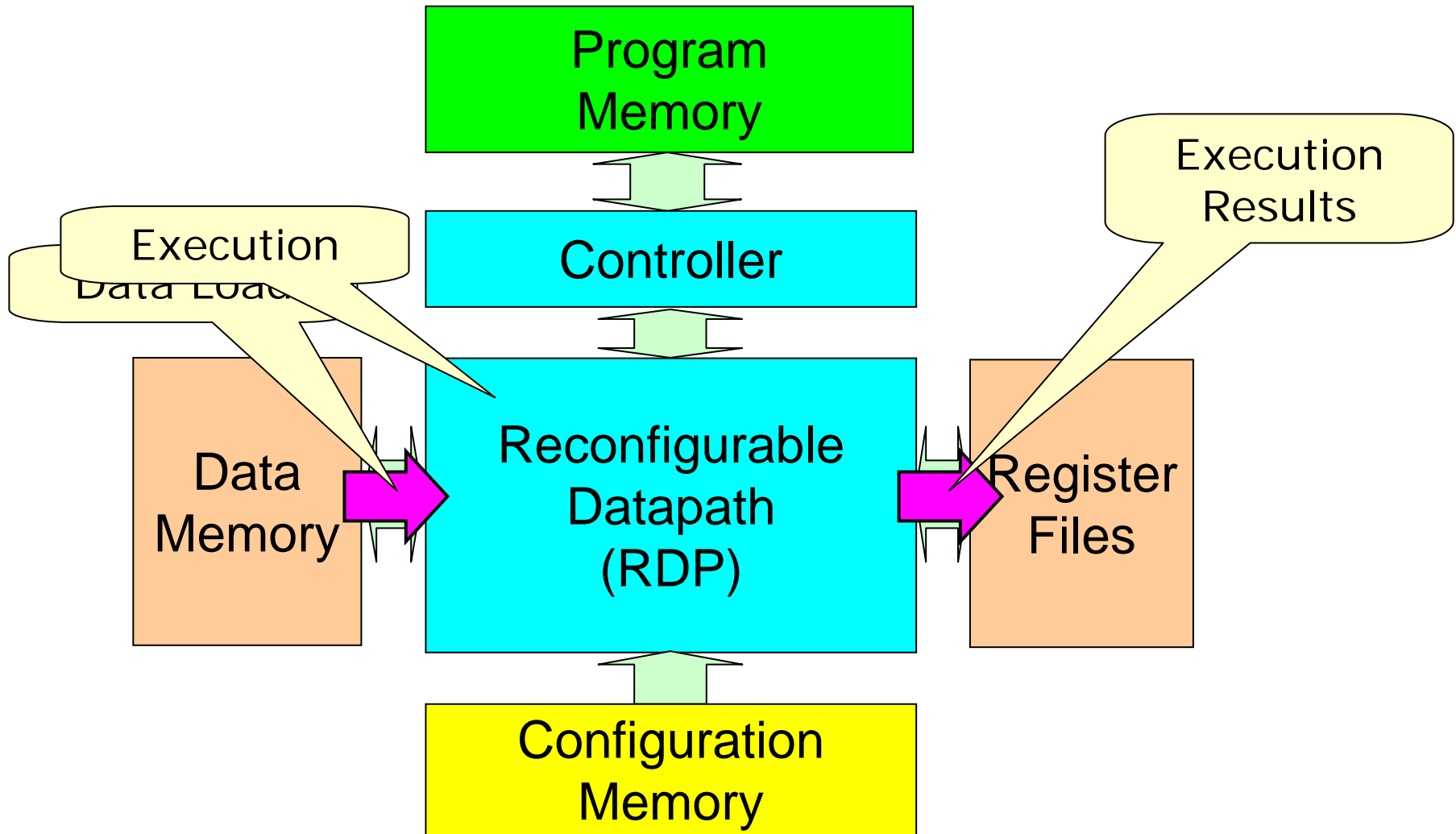
Instruction Execution Flow

- Phase 3: Execution -

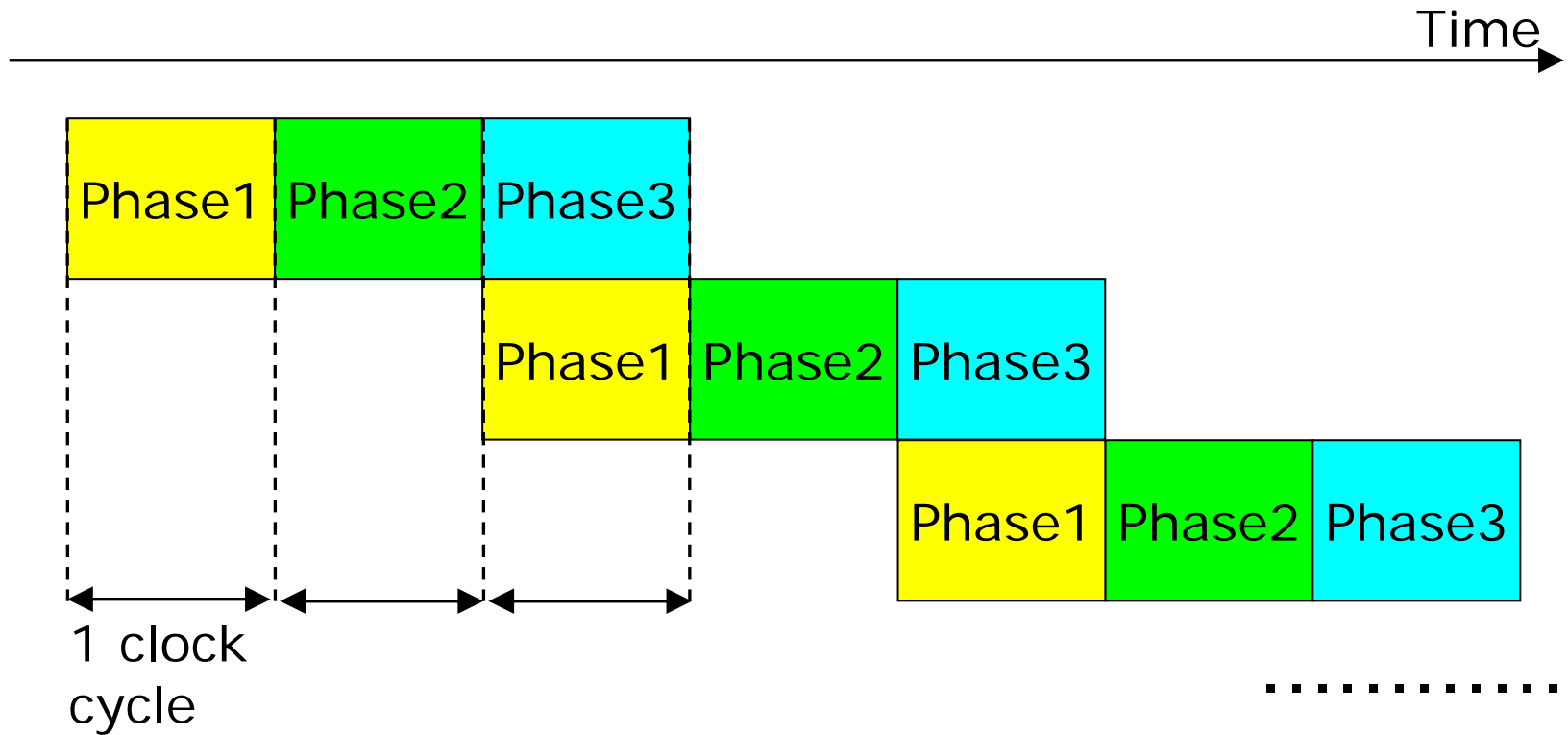


Instruction Execution Flow

- Phase 3: Execution -

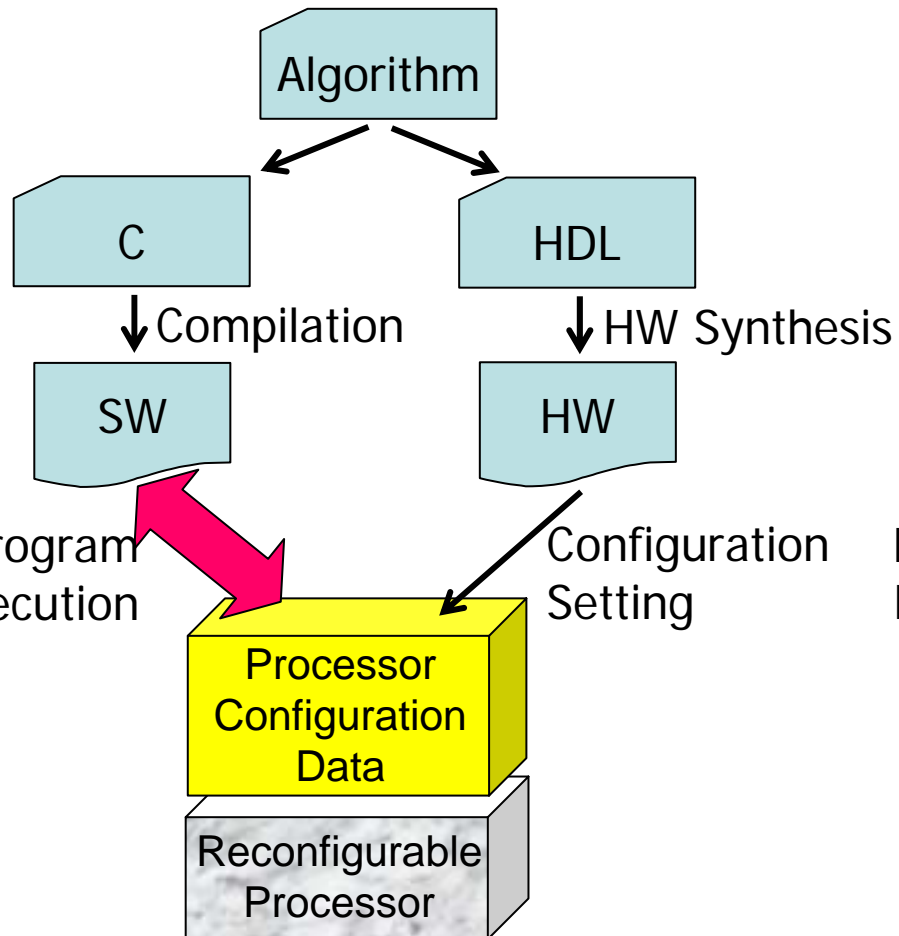


Vulcan Instruction Pipeline

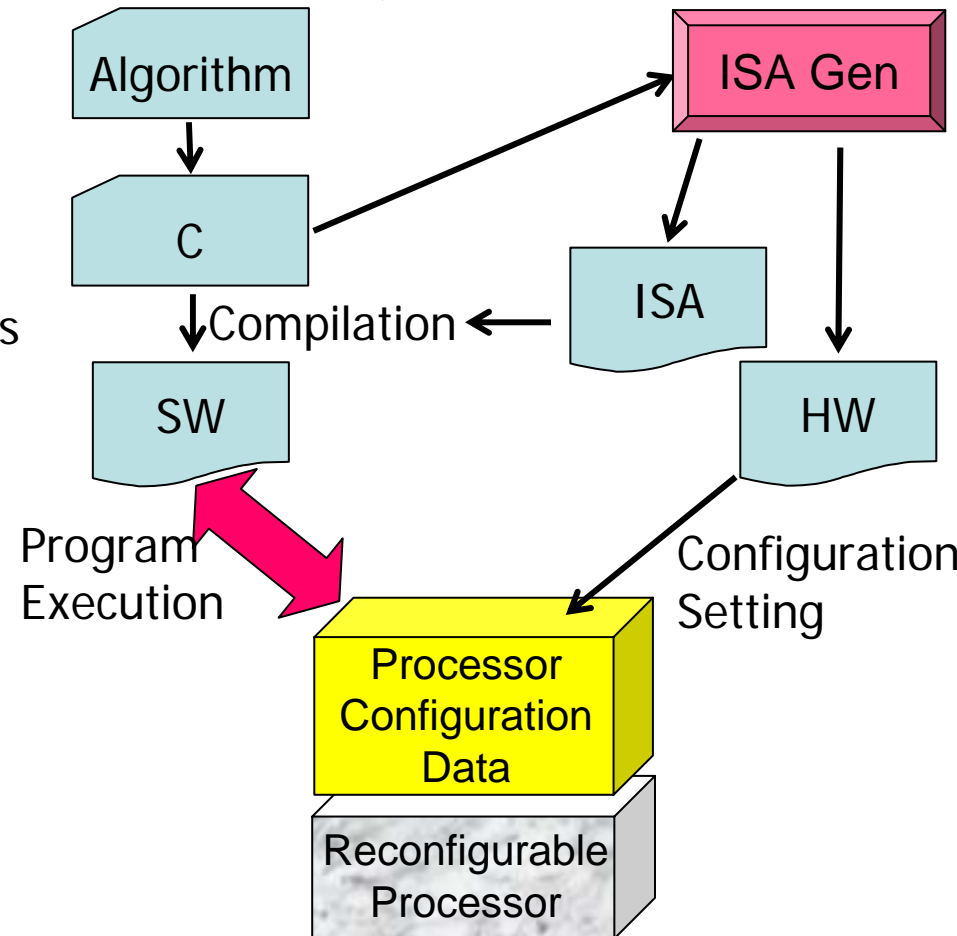


Redefis (Redefinable ISA Processor): A Reconfigurable Processor

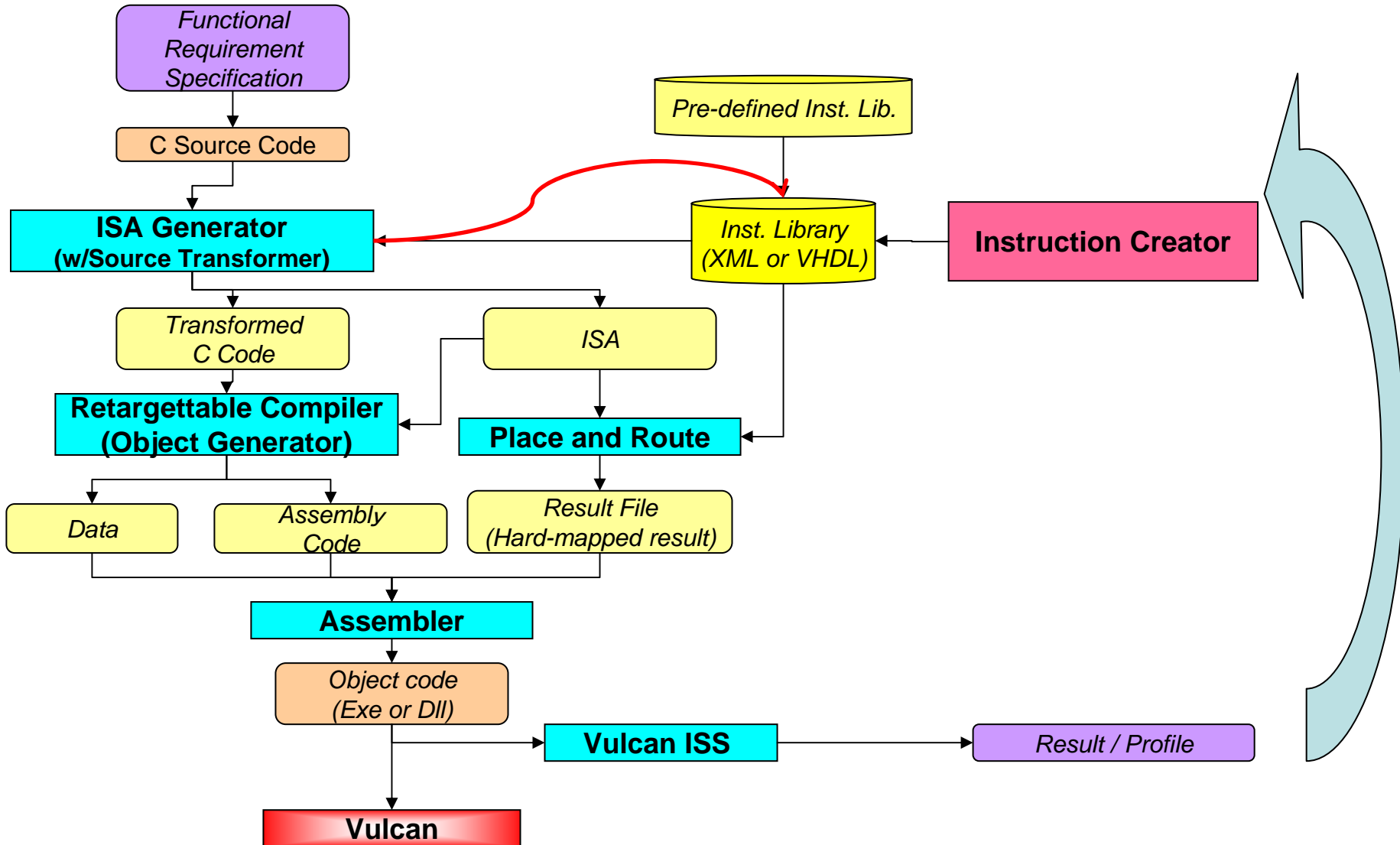
- Normal Reconfigurable Processor



- Redefis (Redefinable ISA Processor)

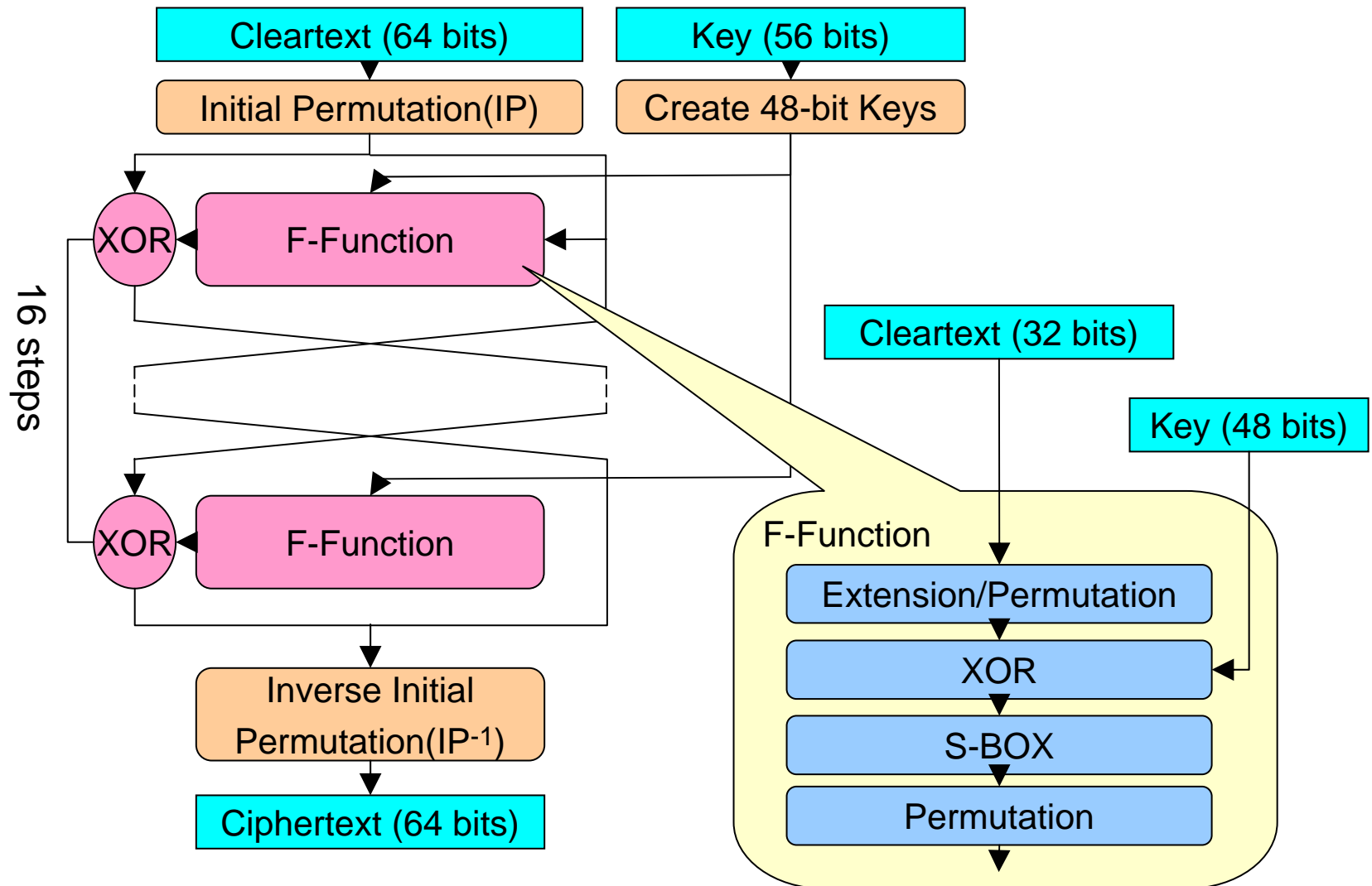


Development Tool Chain

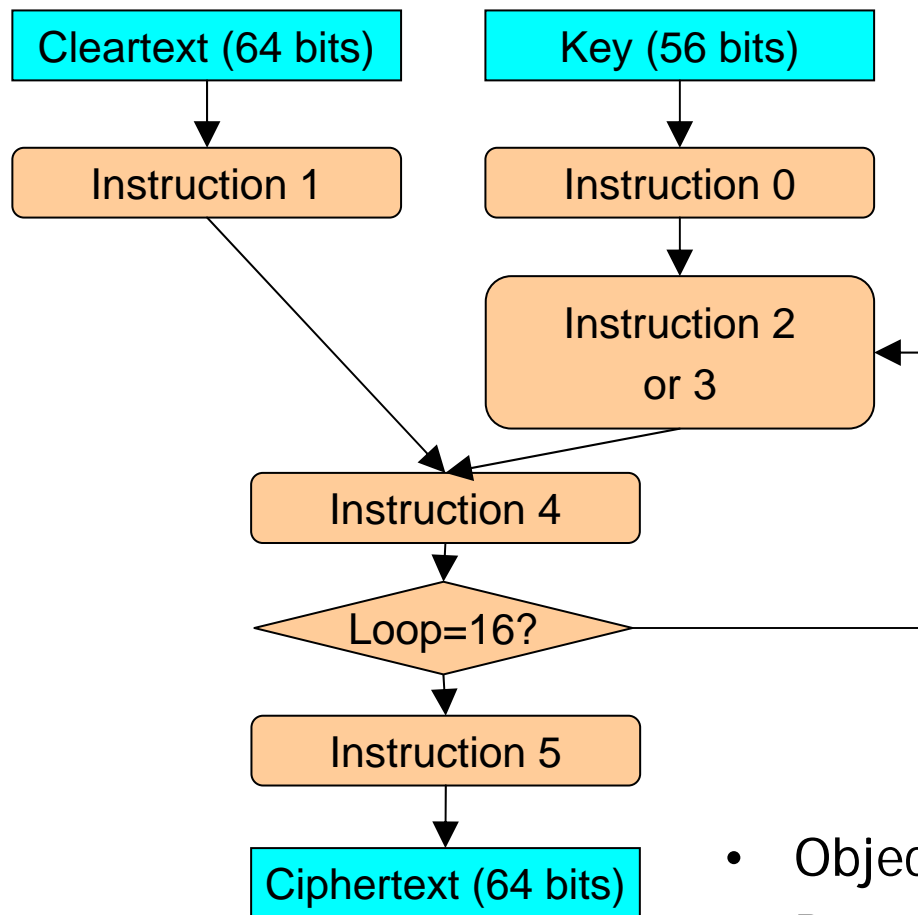


Demo

DES: A Redefis Application



DES: A Redefis Application



| Instruction | What to do |
|-------------|---|
| 0 | Read key and permute it (PC-1) |
| 1 | Read cleartext and permute it (IP) |
| 2 | 1-bit left rotate shift (LS1) |
| 3 | 2-bit left rotate shift (LS2) |
| 4 | Permutation (PC-2), F-function, and XOR |
| 5 | Inverse initial permutation (IP-1), and output ciphertext |

- Object code size: 24 instructions
- Dynamic instruction count: 35 instructions
- Vulcan (6.25MHz) vs. P4 (2.4GHz):
 - Throughput: 570KB/s vs. 150KB/s