A Dynamically Reconfigurable Processor Architecture

Masa Motomura

System ULSI Development Division
NEC Corporation

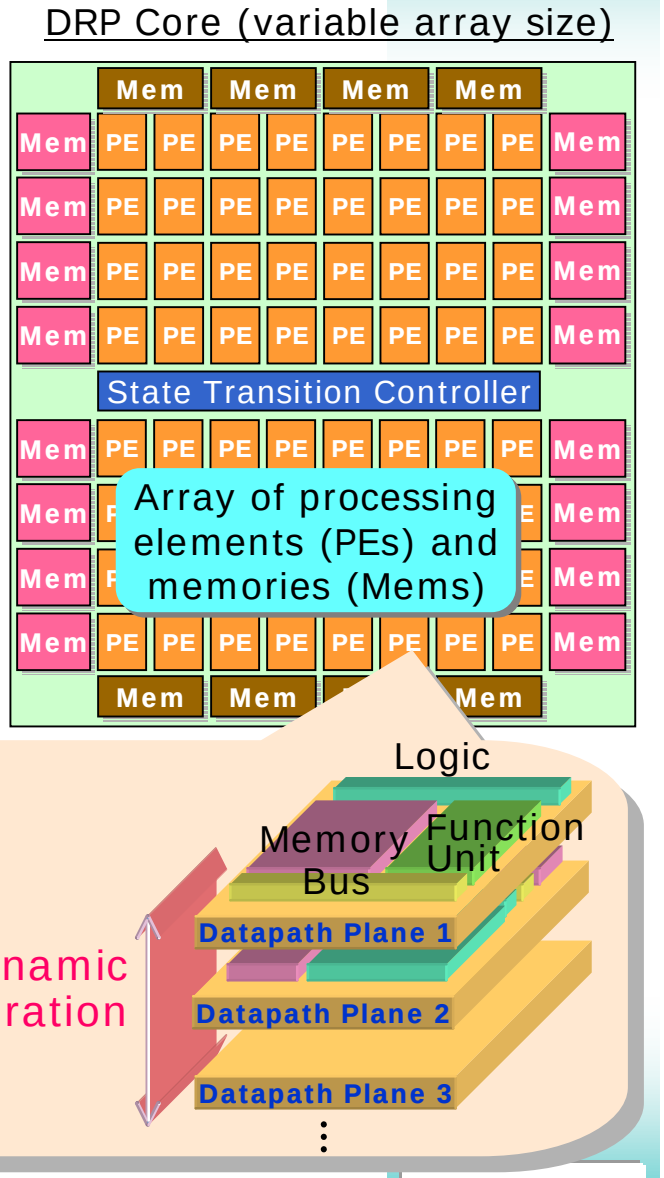
motomura@ah.jp.nec.com

Contents

- Introduction
- DRP architecture
- DRP compiler
- DRP Application development system
- Sample applications
- Future roadmap and summary

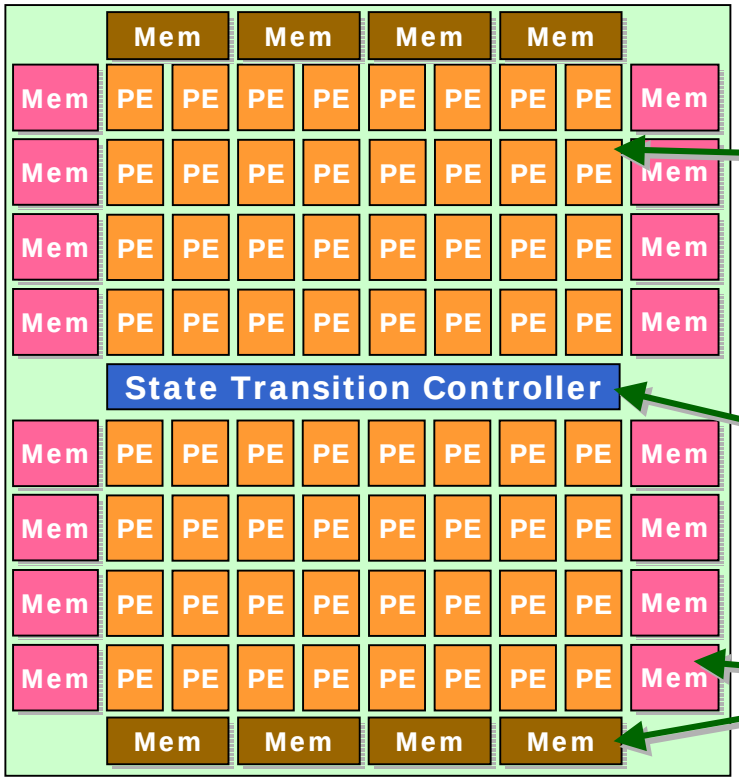
Introduction: What is DRP?

- Dynamically Reconfigurable Processor
  => DRP
- H/W-like performance by customized datapath configurations using array of PEs
- S/W-like scalability by dynamic recon-figuration of customized datapath planes
- C compiler is available
  - Based on in-house high-level synthesis tool, called *Cyber*



DRP Architecture

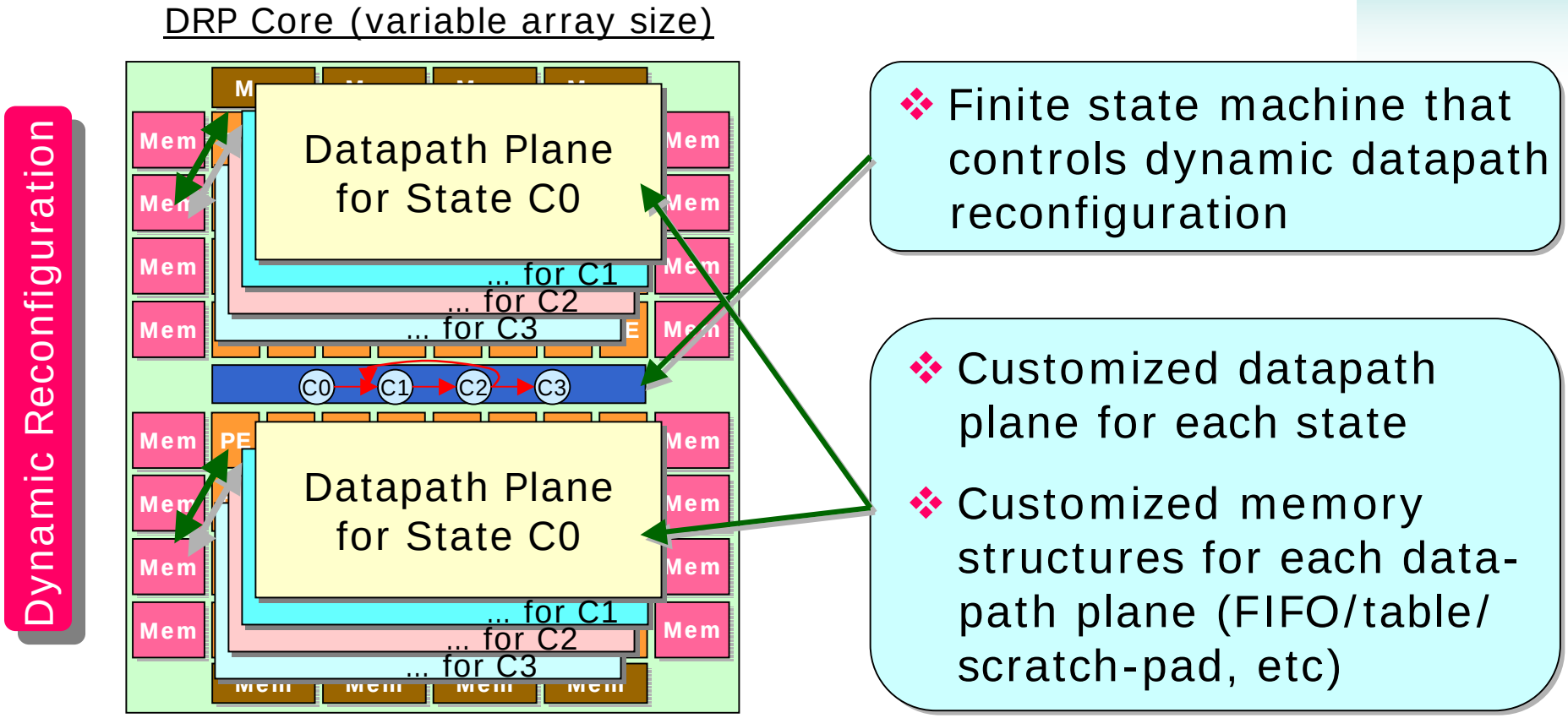
DRP Core (variable array size)



- Array of byte-oriented processing elements
- Fully programmable inter-PE wiring resources
- A simple sequencer
- Array of configurable data memories

- Fine-grained processor-based programmable array architecture
- Architected for stream data processing, such as NW packet, motion/still picture, and wireless data streams, etc.

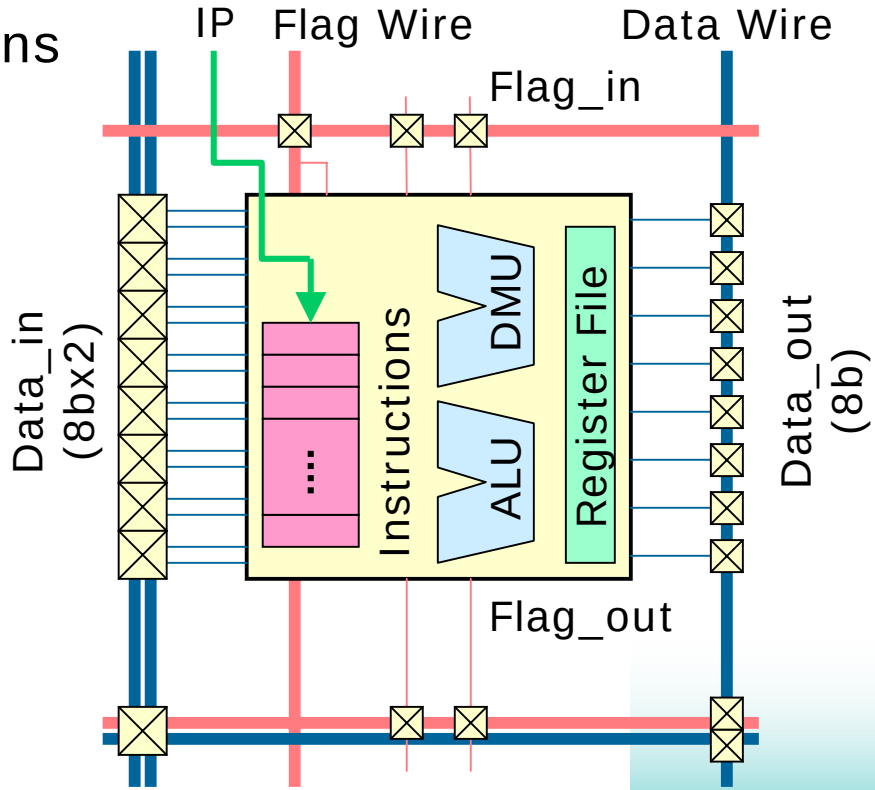
DRP Architecture



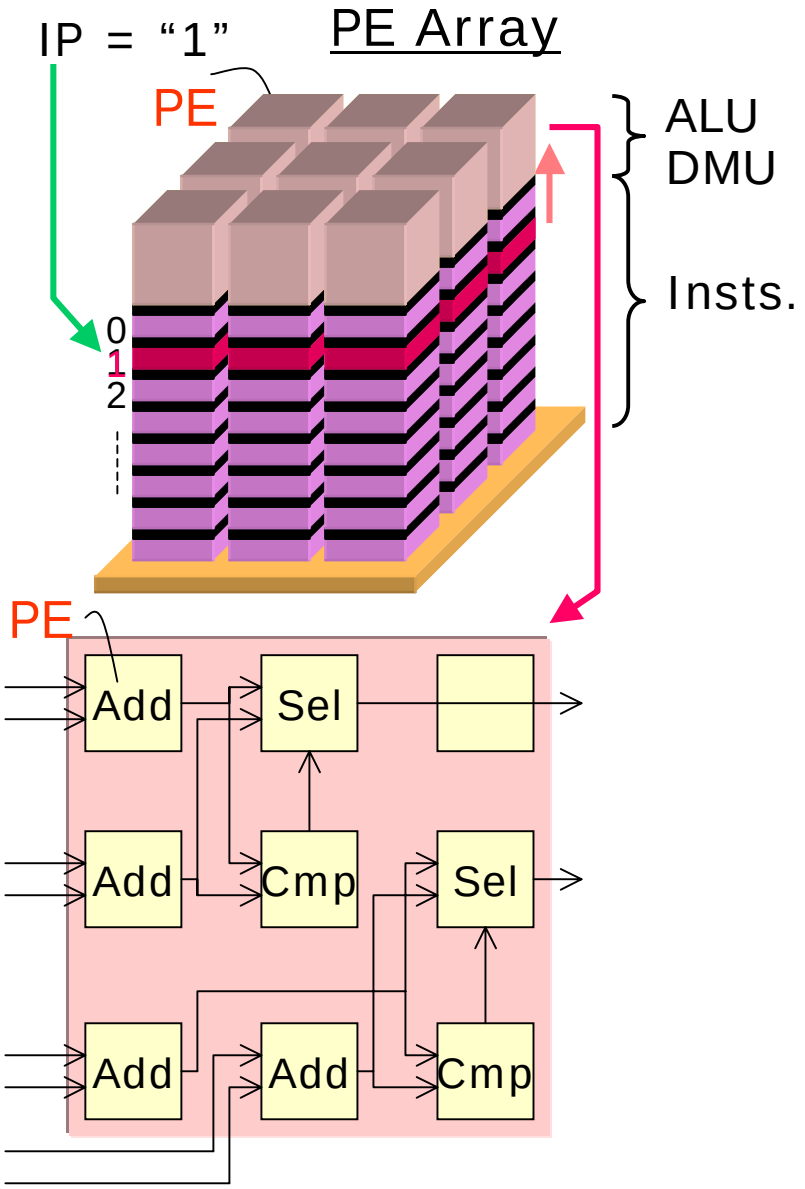
- Sequencer is built-in, and is decoupled from PE array
- Datapath is dynamically reconfigured during runtime
- Customized datapath/memory organization at any time

Processing Element (PE)

- ALU: ordinary byte arithmetic/logic operations
- DMU (data management unit): handles byte select, shift, mask, constant generation, etc., as well as bit manipulations
- An instruction dictates ALU/DMU operations and inter-PE connections
- Source/destination operands can either from/to
  - its own register file
  - other PEs (*i.e.*, flow through)
- Instruction pointer (IP) is provided from STC (state transition controller)

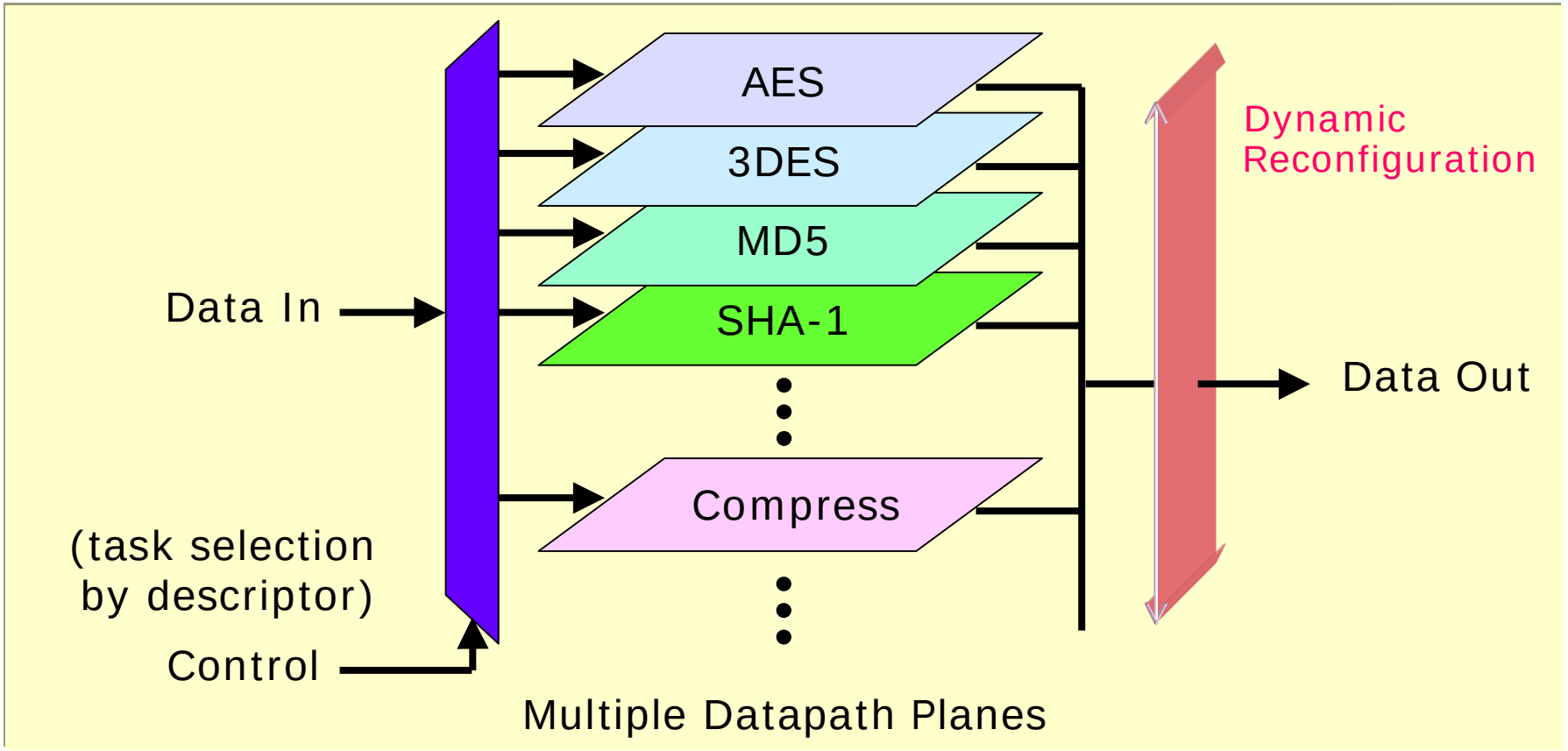


Computation on DRP Core



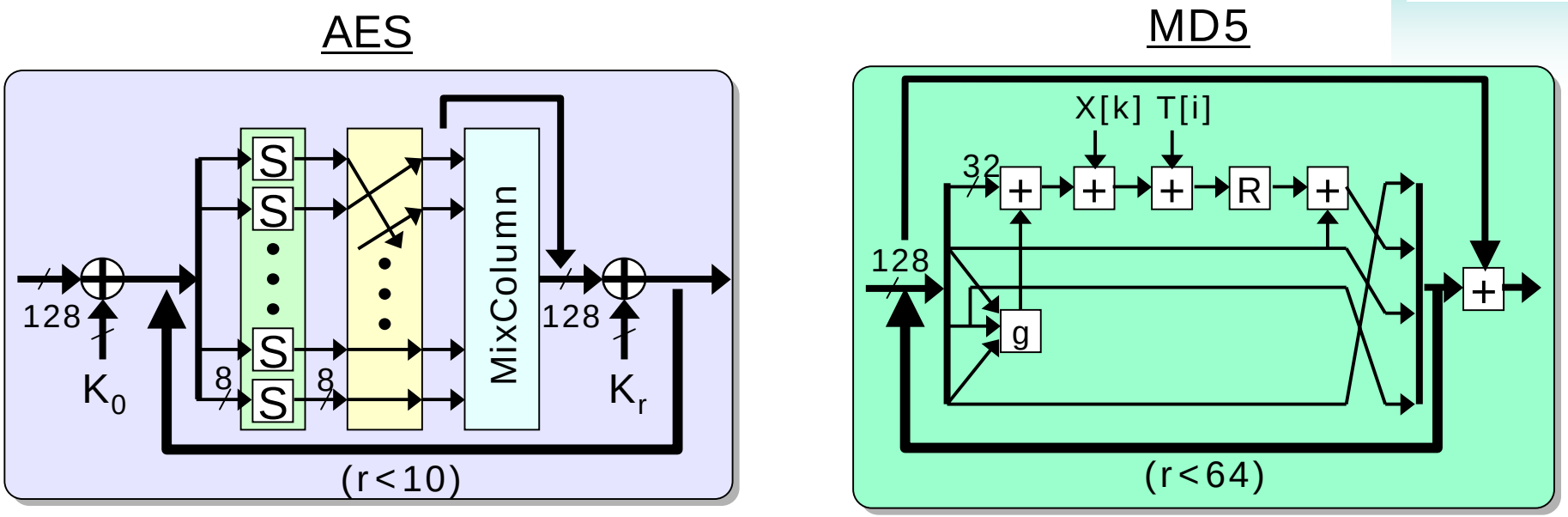
- Instruction Pointer(IP) from STC identifies a datapath plane
- Spatial computation with using a customized datapath plane
- When IP changes, datapath plane switches instantaneously
  - PE instructions as a collection behave like an extreme VLIW
  - Sequencing through instructions => Dynamic reconfiguration

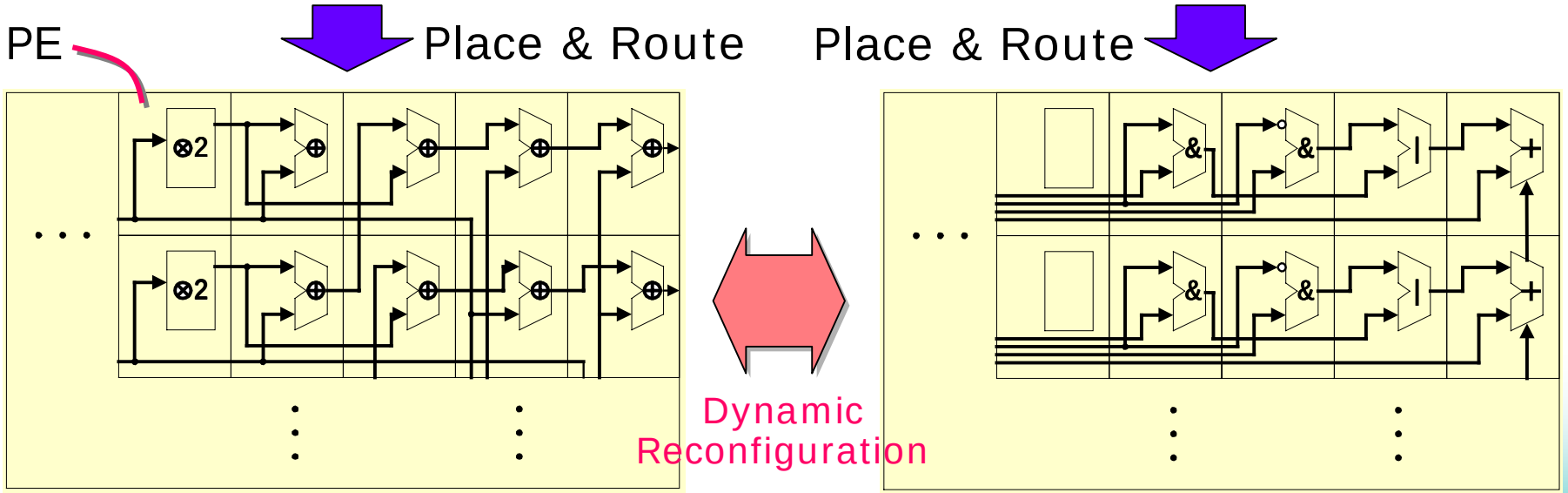
Example: IPSec on DRP Core



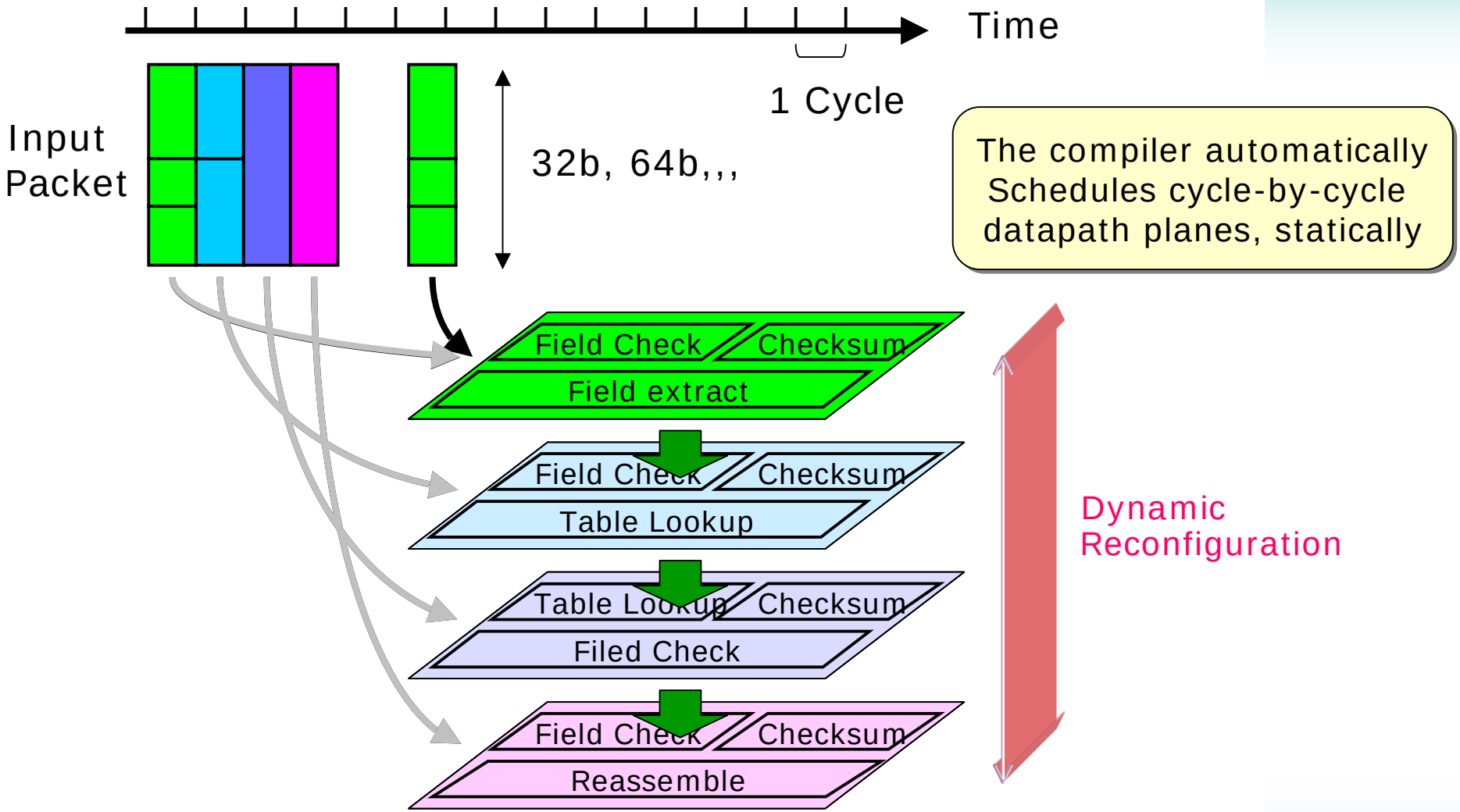
Different datapath plane for different set of algorithmic processing

IPSEC on DRP Core - Continued



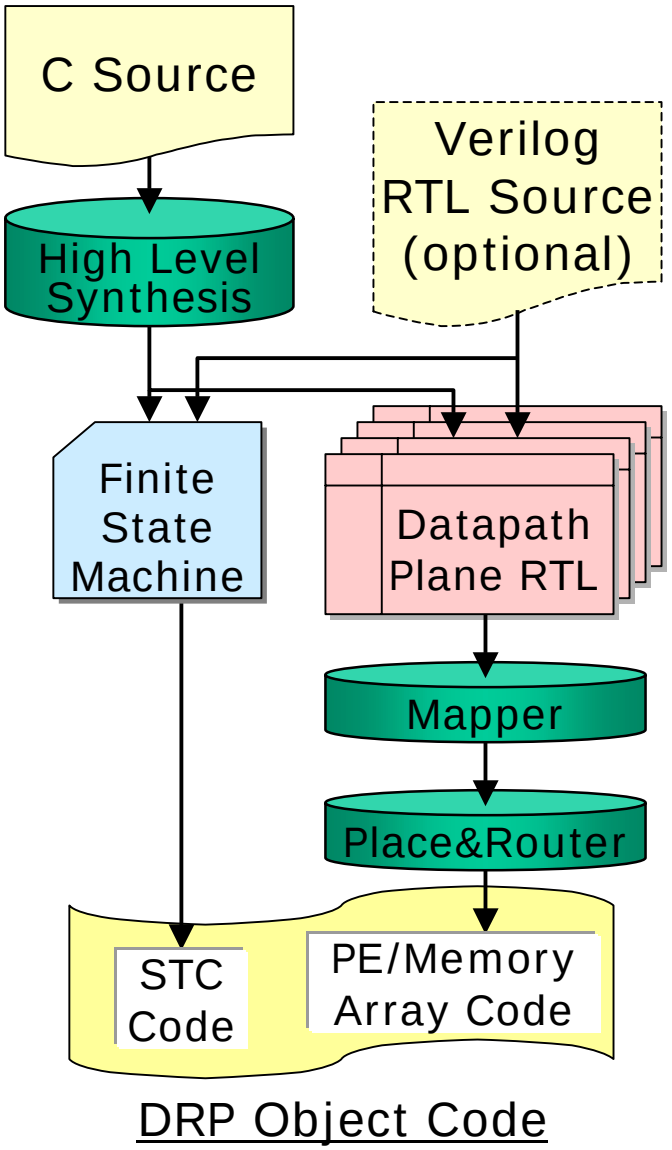


Example: Packet Processing



Different hardware for processing respective chunk of input packet

DRP Compiler



Compile C source code into DRP object code

- High-level synthesis: generates finite state machine and associated datapath planes from the C source code
  - Modify *Cyber* to cope with DRP architectural features
- Mapper: maps RTL for each datapath plane to individual PEs and memories
- Place&Router: physically locate the PEs and memories, and mutually connect them

Programming Model

C source code

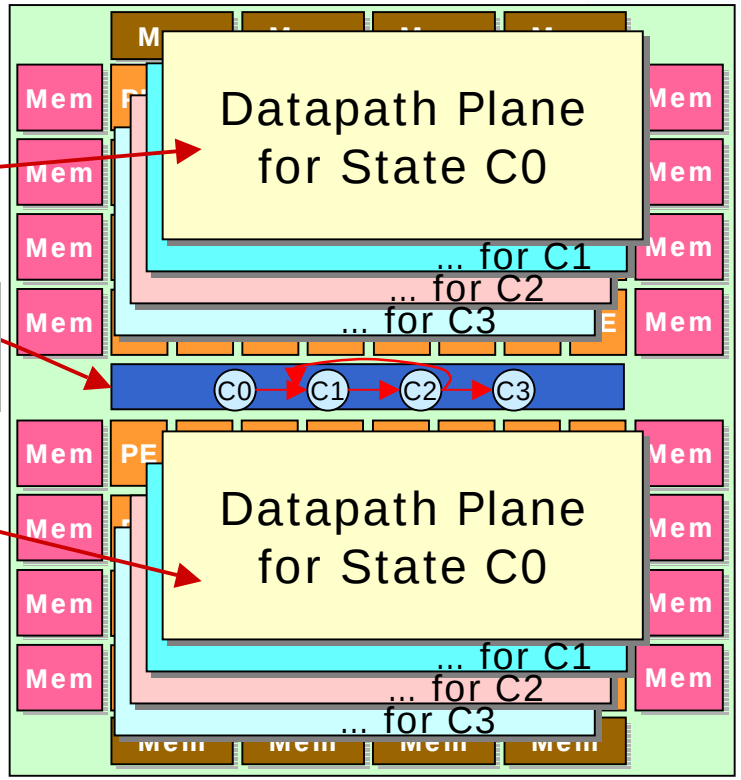
```
if ((a>b)||(c!=d))
    DataPath_Func_1();
else if (e==f)
    DataPath_Func_2();
else
    DataPath_Func_3();
```

Bit-oriented Processing

Control Structure => Finite State Machie

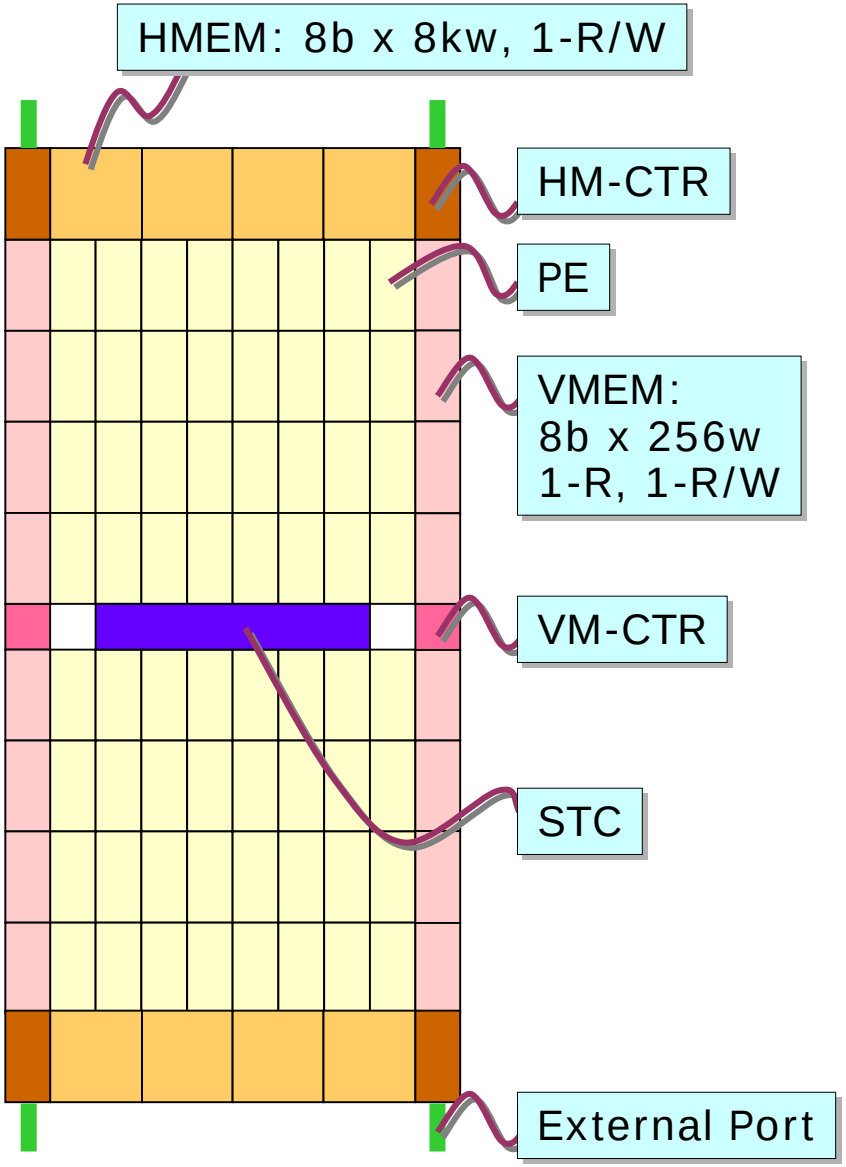
Byte-oriented Processing

DRP-Core



- “Finite state machine with datapath planes”
- DRP is architected based on a clear picture of how C code is compiled into hardware

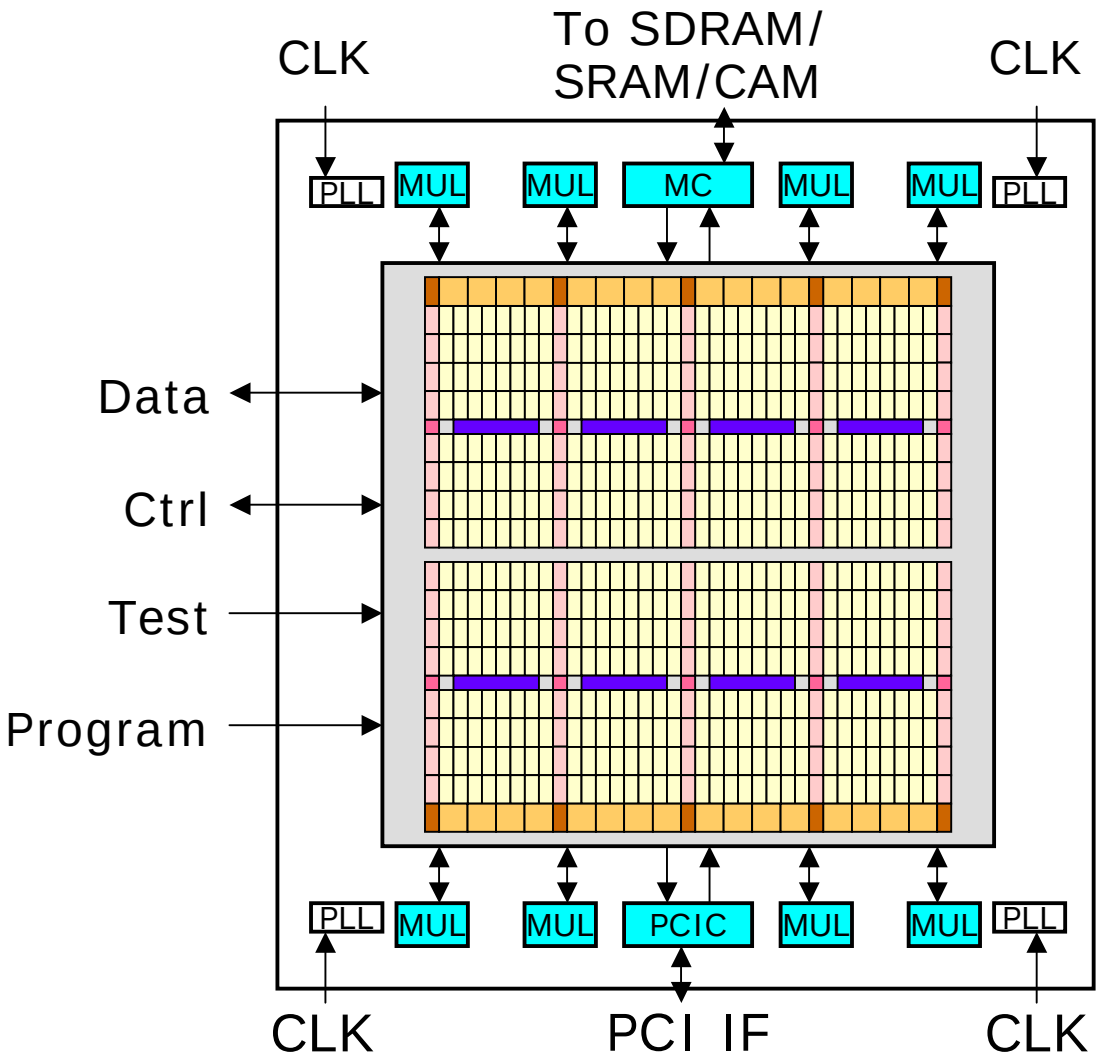
1st Implementation: DRP Tile



- Tile is a minimum unit of DRP array
  - 8x8 PEs
  - 1-p HMEM and 2-p VMEM
- A PE stores 16 instructions
- Reconfiguration of a tile is controlled by its STC
- External port can attach extended functional units, such as
  - External memory controllers
  - Peripheral bus controllers (like PCI)
  - Complex operation units (like multiplier, divider, etc)

1st Silicon: DRP-1

DRP-1 Chip Blockdiagram



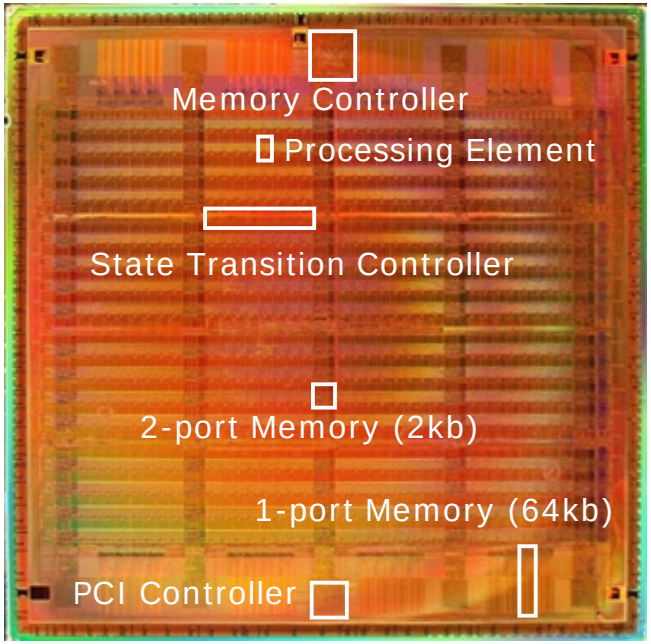
- 8 Tiles. 512 PEs, 160kb VMEMs, 2Mb HMEMs
- 8 STCs
- 8 32b Multipliers
- 1 SDRAM/SRAM/CAM controller
- 1 PCI controller
- 4 PLLs

- PE
- VMEM
- HMEM
- VM-CTR
- HM-CTR
- STC
- Extended Functional Units

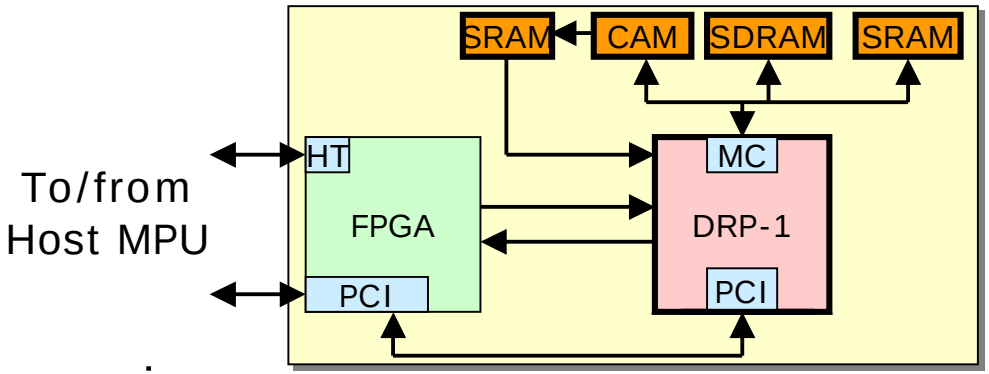
Current Status: HW Platform

NEC

DRP-1 Die Photo

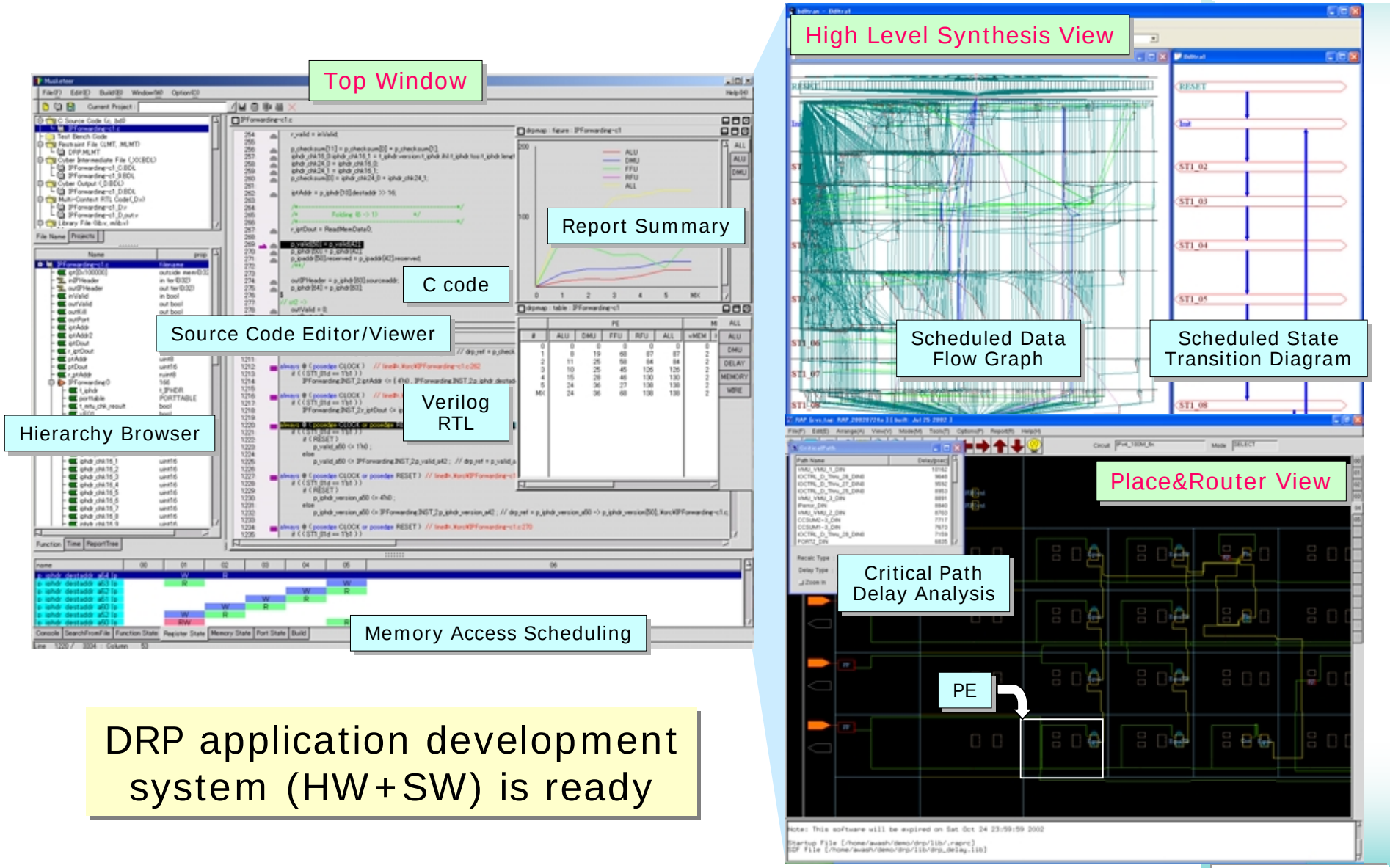


- 0.15µm CMOS 8-Al process, 696-pin TBGA package
- 33-133MHz clock
- 22M logic transistors (44K for a PE)
- 1.5Mb configuration memories
- Fully functional w/o respin



- DRP-1 board is now up and running
  - Demonstration of DRP as an off-loading engine to a host MPU through PCI or HyperTransport I/F

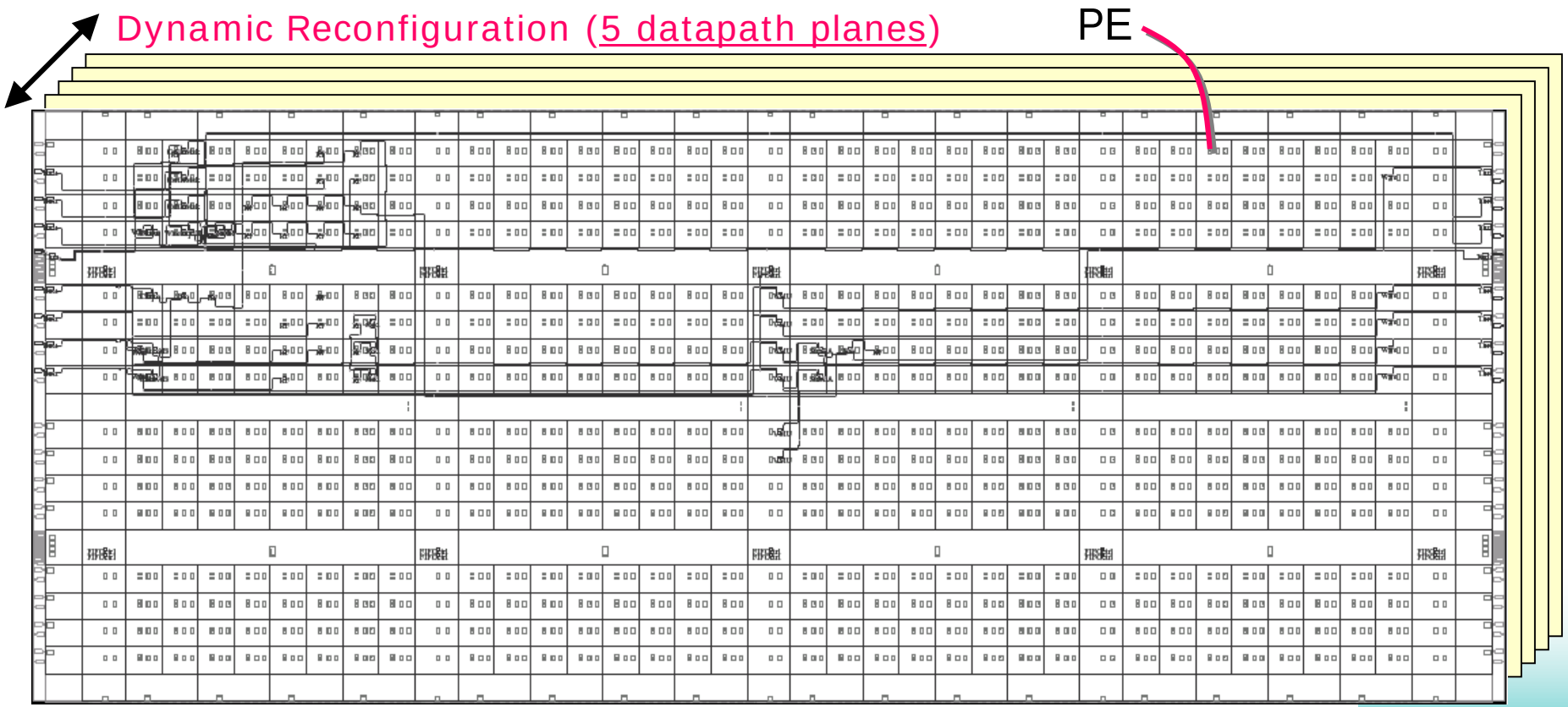
Current Status: Compiler Suite



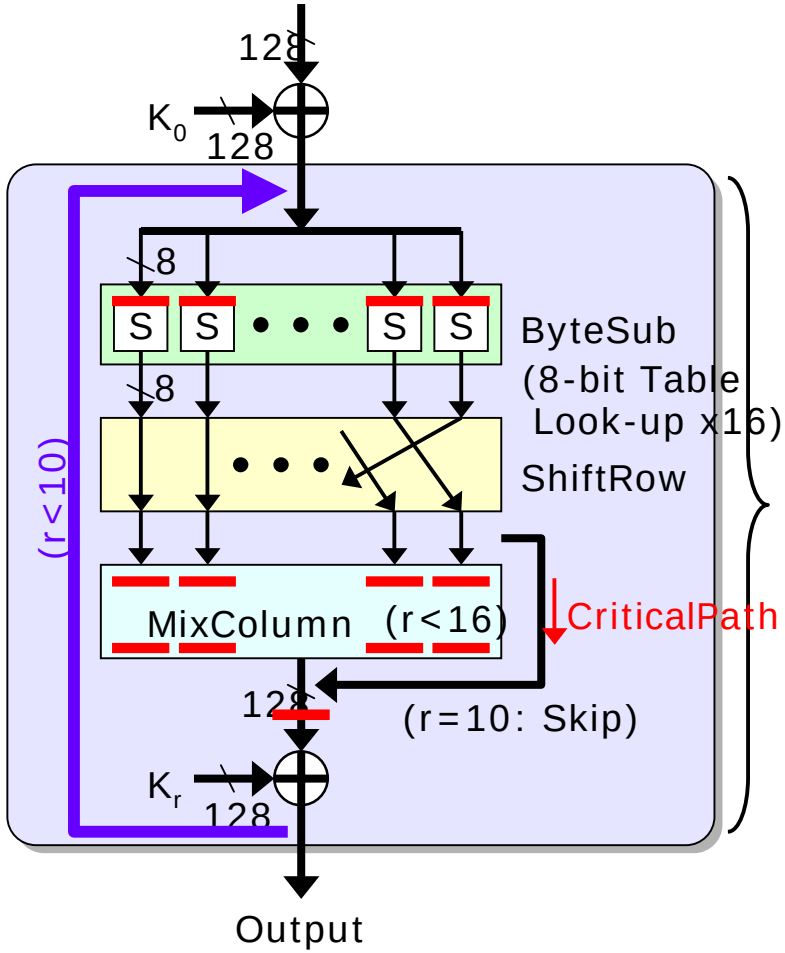
DRP application development system (HW+SW) is ready

Example: IPv4 Packet Forwarding

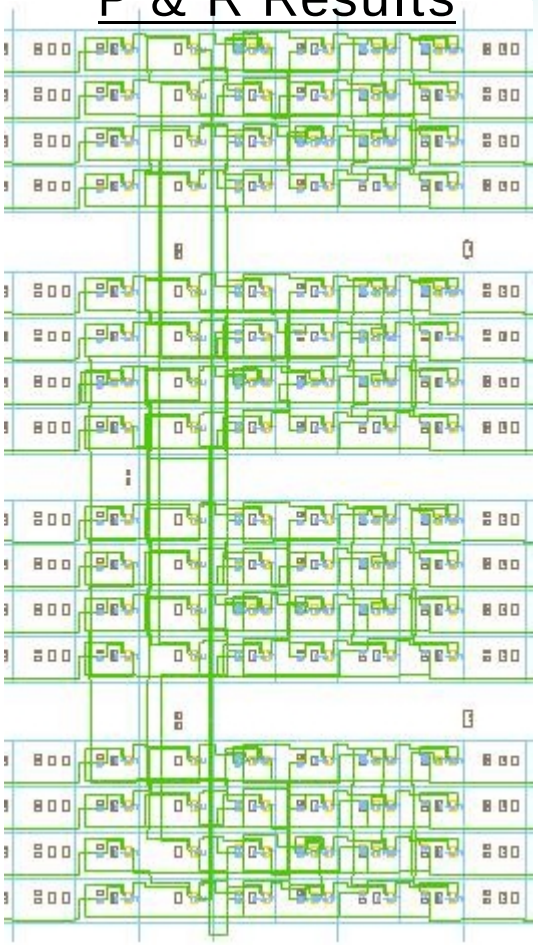
- RFC1812 (Header check, checksum, longest prefix match, header modification)
- 8x16 PEs, 100MHz Operation
- Table look up: 5 round index search using 100MHz SRAM
- 6 pipe-stages, 5-cycle each. 100MHz x 5Cycle => 20M Packet/second



Example: AES Encryption



P & R Results



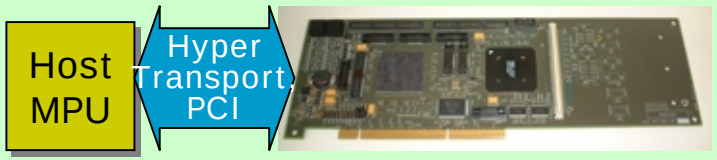
- Critical Path: 6.6ns. Throughput: 1.76Gbps (ECB mode)
- Fits in 8x16 PEs, single context (80 ALU/DMUs, 96 RFUs, 16vMEMs)

DRP Roadmap

NEC

DRP-based Product

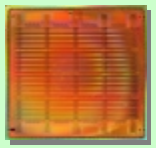
Application development system



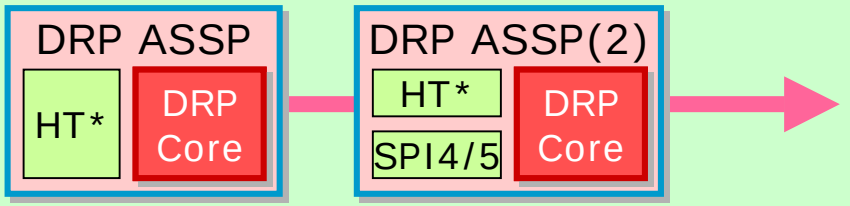
Embedded core for NEC ASICs

- Faster time to market
- Longer time in market
- Lower risk ASIC design

DRP 1st Silicon

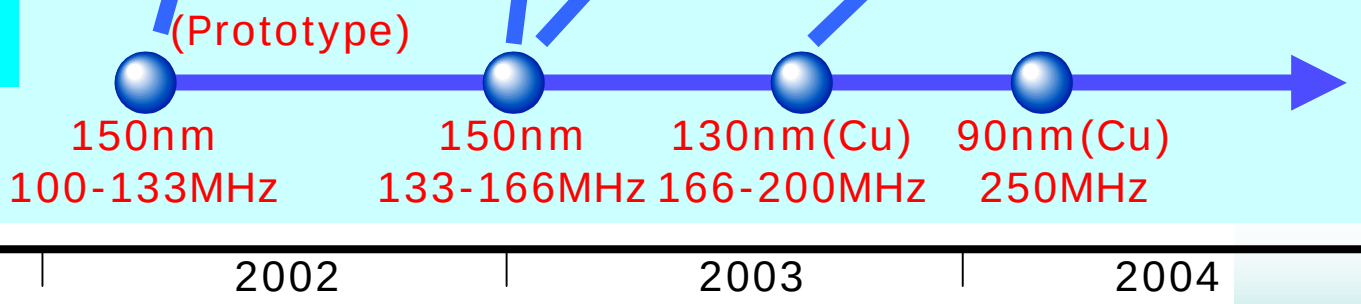


Programmable off-load engine to host MPU



* PCI-X/Express may also be available

DRP Core



Summary

- DRP architecture features
  - Array of PEs and memories for programmable datapath
  - FSM-based sequencer, a state transition controller, for the control of dynamic datapath reconfiguration
- DRP application development system (DRP-1 chip & board, and the compiler suite) is ready
- Sample application development is underway
- Launch of DRP-based product planned during Y'03