

DAPDNA-2

A Dynamically Reconfigurable Processor with 376 32-bit Processing Elements

Tomoyoshi Sato
Vice President & CTO
IPFlex Inc.

IP FLEX® Agenda

- **Overview**
- **Design Goals and Decisions**
- **Overall Architecture**
- **Processing Element (PE) Architecture**
- **Interconnect Architecture**
- **Application Construction**
- **Performance**
- **Advanced Usages**
- **Summary**

IP FLEX® DAPDNA-2

- 32bit RISC + Reconfigurable Fabric + Peripherals
 - Fujitsu 0.11 μ m 7Cu+1Al
 - 12 M gates
 - 1156-pin FCBGA, 2.4V I/O, 1.2V Core
 - 166 MHz, 3-7 W
-
- Suited for stream processing
 - 10-50x performance of 3GHz general-purpose CPU





Design Goals and Decisions

- **High performance**
 - Massively parallel processing elements
- **Flexibility**
 - Field programmable
- **Versatility**
 - Dynamically reconfigurable with small overhead
- **Ease of use**
 - Fixed-frequency coarse-grained ALUs
- **Scalability**
 - High-bandwidth I/O (interconnect) channels



DAPDNA-2 Architecture

- **CPU for sequential tasks**

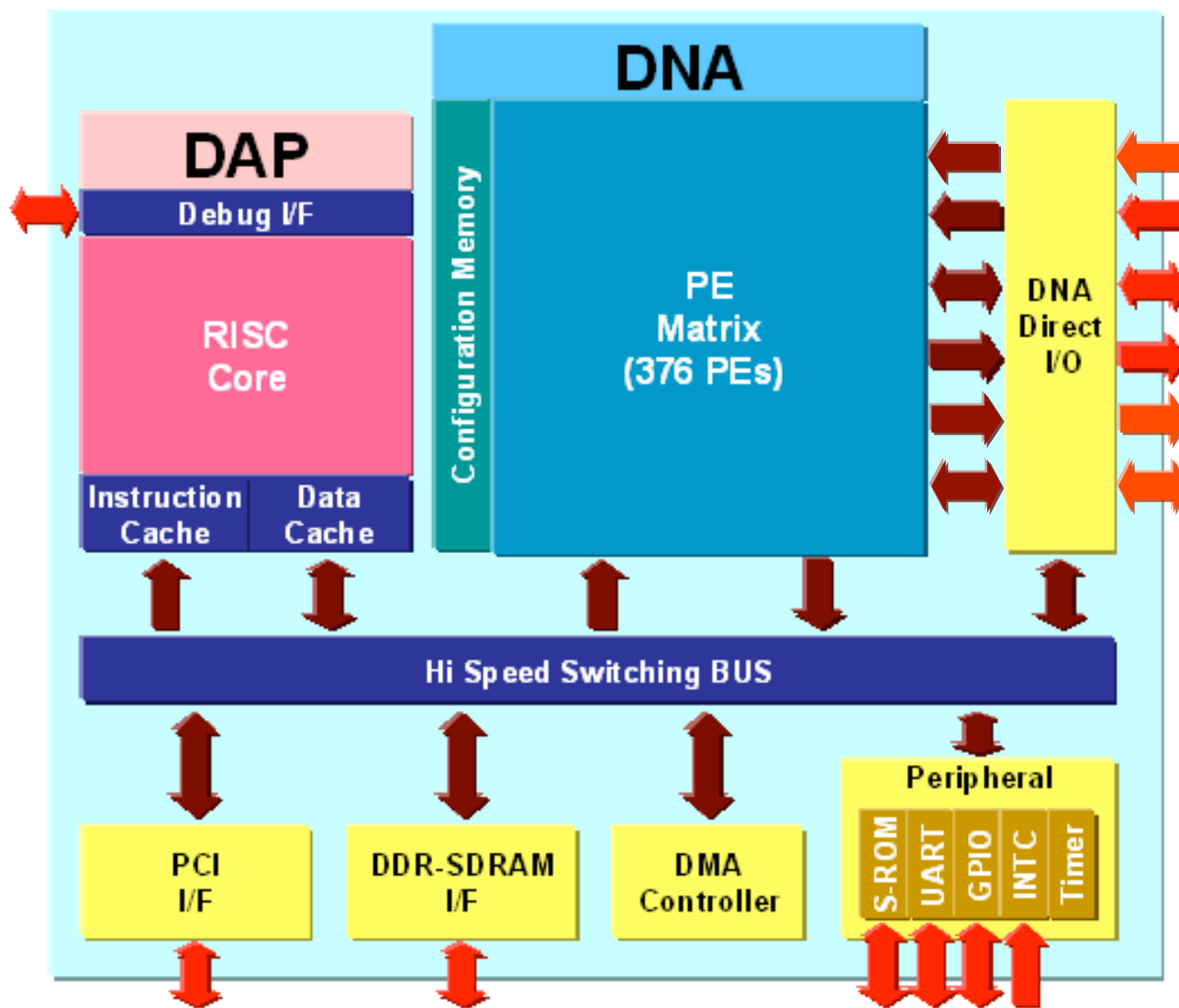
- DAP (Digital Application Processor)
- 32-bit RISC with 2way 8k+8k I/D caches

- **Reconfigurable fabric for parallel processing**

- DNA (Distributed Network Architecture)
 - 376 heterogeneous 32-bit processing elements
 - ALUs, RAMs, delays, counters, I/O buffers
 - 4 configuration banks: switchable in one cycle
 - 28 billion ALU operations / s (166MHz x 168)
 - 9 billion 16x16 multiplications / s (166MHz x 56)
- 2.66 GB/s memory bandwidth (64-bit DDR166)
- 4 GB/s I/O bandwidth (32-bit x 166MHz x 6ch)



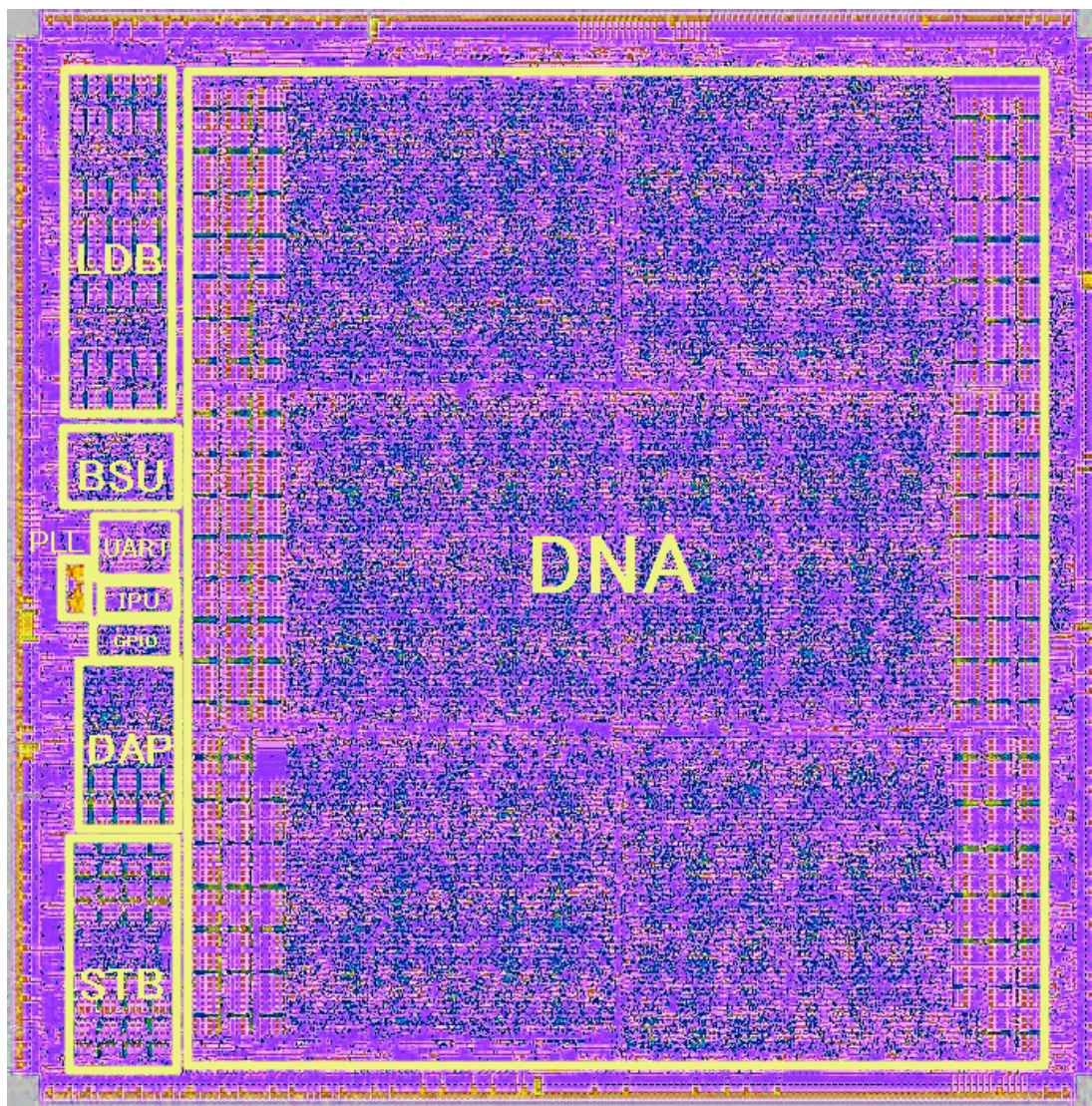
DAPDNA-2 Block Diagram



www.DataSheet4U.com

IP FLEX® DAPDNA-2 Chip

www.DataSheet4U.com





DAPDNA-2 Reconfigurable Fabric

		0	1	2	3	4	5	6	7			0	1	2	3	4	5	6	7			
Segment 0	0	LDB		LDB		LDB		LDB		EXC	EXC	EXF	EXF	LDX	LDX	LDX	LDX			0		
	1	RAM	C16L	C16L	C16L	C16L	C16E	C16E	DLH	DLH	C32L	C32L	C32L	C32L	C32E	C32E	C32E	RAM			1	
	2	RAM	EXC	EXC	EXC	EXC	EXC	EXC	EXF	DLH	DLH	EXF	EXC	EXC	EXC	EXC	EXC	EXC	RAM			2
	3	RAM	EXS	EXS	EXS	EXS	EXS	EXS	EXF	DLH	DLH	EXF	EXS	EXS	EXS	EXS	EXS	EXS	RAM			3
	4	RAM	EXR	EXR	EXR	EXR	EXR	EXR	EXF	DLH	DLH	EXF	EXR	EXR	EXR	EXR	EXR	EXR	RAM			4
	5	RAM	DLE	DLE	DLE	DLE	DLE	DLE	DLE	DLH	DLH	DLE	DLE	DLE	DLE	DLE	DLE	DLE	RAM			5
	6	EXM	EXM	EXM	EXM	EXM	EXM	EXM	EXM	DLH	DLH	EXM	EXM	EXM	EXM	EXM	EXM	EXM	EXM			6
	7	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLX	DLX	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLV			7
Segment 1	0	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLX	DLX	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLX			0	
	1	RAM	EXM	EXM	EXM	EXM	EXM	EXM	DLH	DLH	EXM	EXM	EXM	EXM	EXM	EXM	EXM	RAM			1	
	2	RAM	EXF	EXF	EXF	EXF	EXS	EXM	DLH	DLH	EXM	EXS	EXF	EXF	EXF	EXF	EXF	RAM			2	
	3	RAM	DLE	DLE	DLE	DLE	DLE	DLE	DLH	DLH	DLE	DLE	DLE	DLE	DLE	DLE	DLE	RAM			3	
	4	RAM	EXC	EXC	EXC	EXC	EXS	EXS	DLH	DLH	EXS	EXS	EXC	EXC	EXC	EXC	EXC	RAM			4	
	5	RAM	EXR	EXR	EXR	EXR	EXS	EXM	DLH	DLH	EXM	EXS	EXR	EXR	EXR	EXR	EXR	RAM			5	
	6	RAM	EXM	EXM	EXM	EXM	EXM	EXM	DLH	DLH	EXM	EXM	EXM	EXM	EXM	EXM	EXM	RAM			6	
	7	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLX	DLX	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLV			7	
Segment 2	0	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLX	DLX	DLV	DLV	DLV	DLV	DLV	DLV	DLV	DLV			0	
	1	EXM	EXM	EXM	EXM	EXM	EXM	EXM	DLH	DLH	EXM	EXM	EXM	EXM	EXM	EXM	EXM	EXM			1	
	2	RAM	DLE	DLE	DLE	DLE	DLE	DLE	DLH	DLH	DLE	DLE	DLE	DLE	DLE	DLE	DLE	RAM			2	
	3	RAM	EXR	EXR	EXR	EXR	EXR	EXR	EXF	DLH	DLH	EXF	EXR	EXR	EXR	EXR	EXR	RAM			3	
	4	RAM	EXS	EXS	EXS	EXS	EXS	EXS	EXF	DLH	DLH	EXF	EXS	EXS	EXS	EXS	EXS	RAM			4	
	5	RAM	EXC	EXC	EXC	EXC	EXC	EXC	EXF	DLH	DLH	EXF	EXC	EXC	EXC	EXC	EXC	RAM			5	
	6	RAM	C16S	C16S	C16S	C16S	C16E	C16E	DLH	DLH	C32S	C32S	C32S	C32S	C32S	C32E	C32E	RAM			6	
	7	STB		STB		STB		STB		EXC	EXC	EXF	EXF	STX	STX	STX	STX			7		
																				Segment 3		
																				Segment 4		
																				Segment 5		

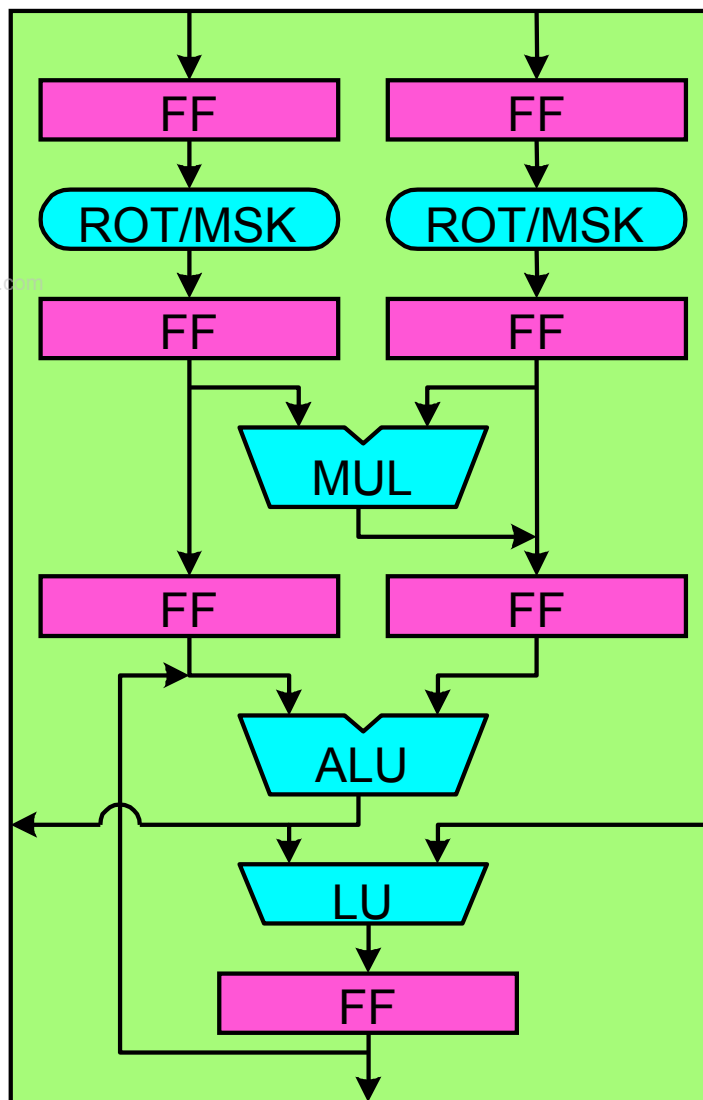
www.DataSheet4U.com



Processing Elements

Type	Qty.	Description / Example
ALU	168	EXM : ALU with 16x16 -> 32 multiplier EXS : ALU with byte-swap instruction EXC, EXR, EXF :
RAM	32	16kB, 8/16/32-bit data width
delay	136	DLE : programmable (1-13) delay DLV : vertical cross-segment delay DLH, DLX :
counter	24	C16L : 16bit address counter for load C16S, C16E, C32L, C32S, C32E :
I/O buffer	16	LDB : load buffer from external memory LDX : load buffer from I/O channel STB, STX :

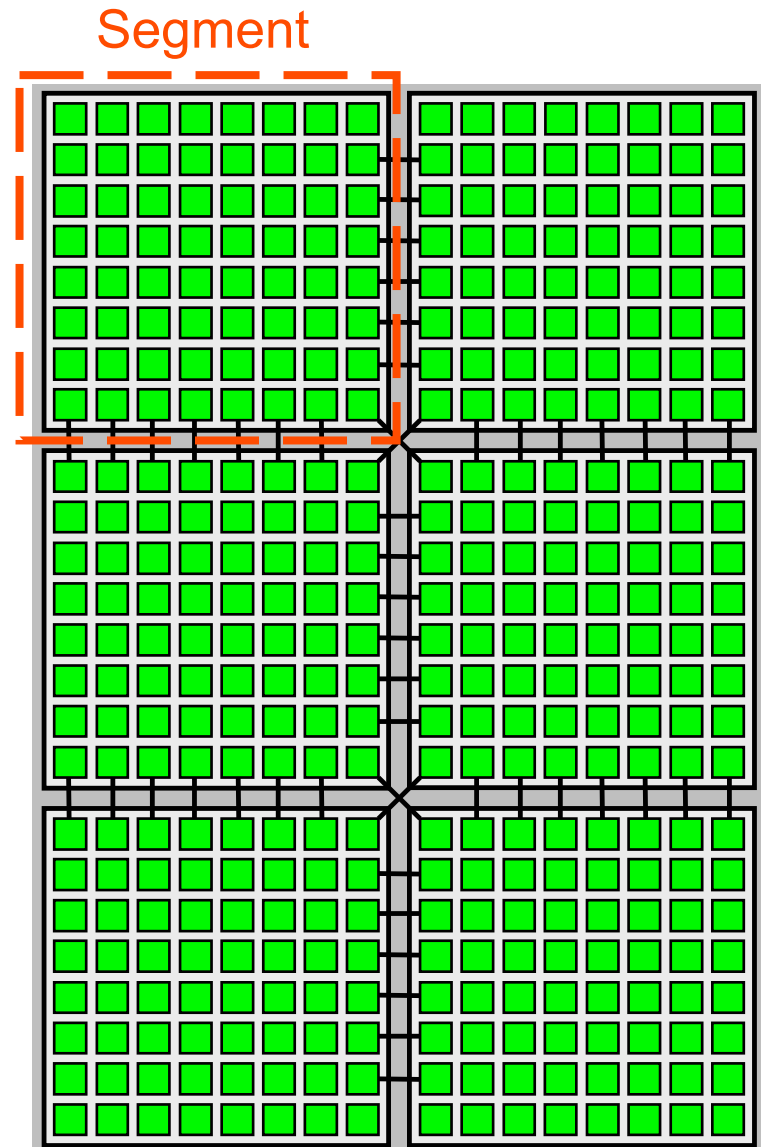
PE Example : ALU with 16x16 MUL



- **Finely pipelined**
 - Guarantees 166-MHz operation
- **Registered input/output**
 - Gives sufficient time to interconnect wires
- **ALU feedback path**
 - Supports accumulation in one-cycle throughput
- **Stage bypass path (not shown)**
 - Reduces latency when some stages are unused

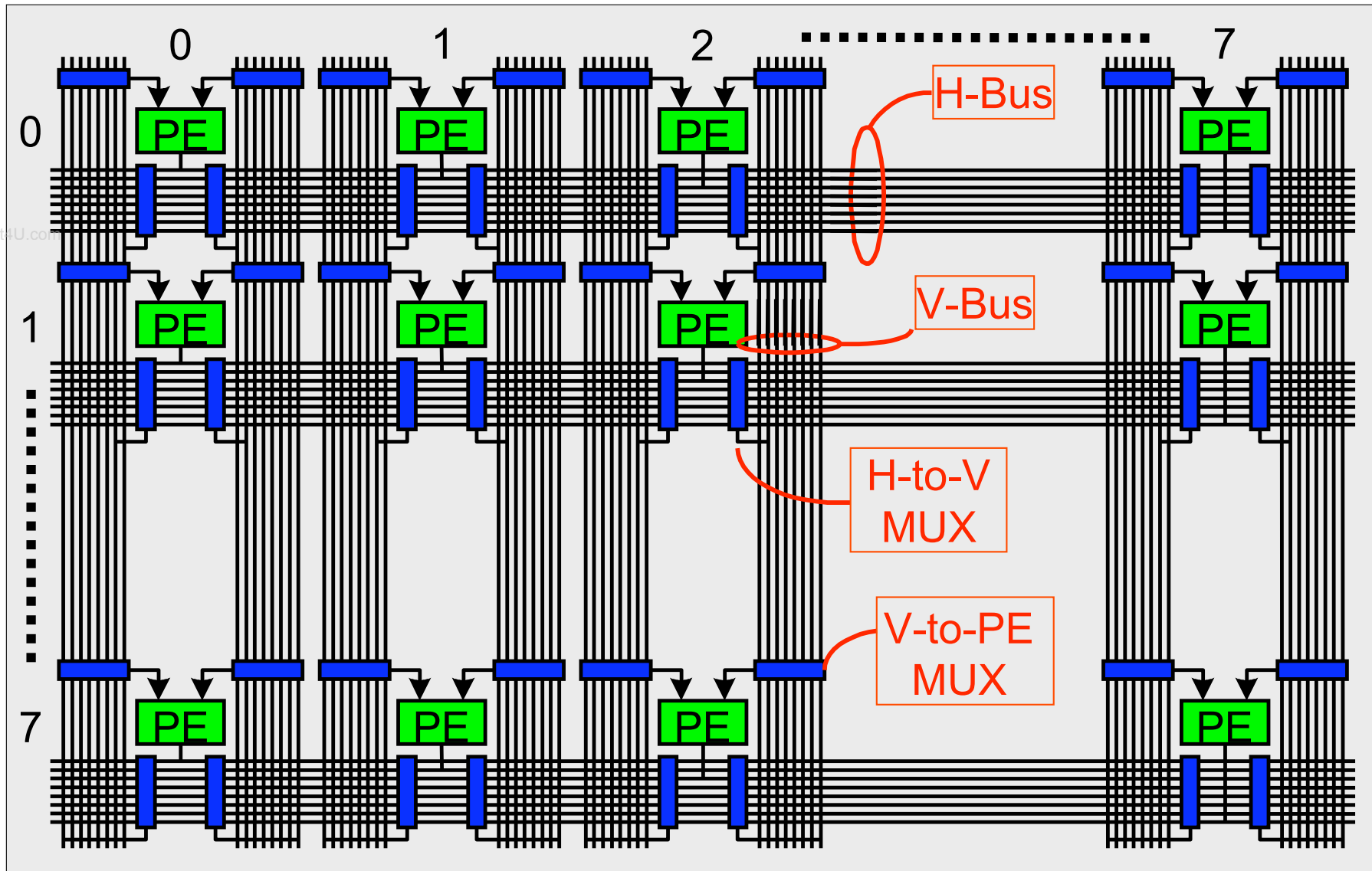
Segmented Interconnect

- **8x8 PEs in 1 segment**
- **1-cycle intra-segment interconnect**
 - Nearly freely routable
- **Inter-segment routing via delay elements on segment boundaries**
 - Nearest-neighbor connection



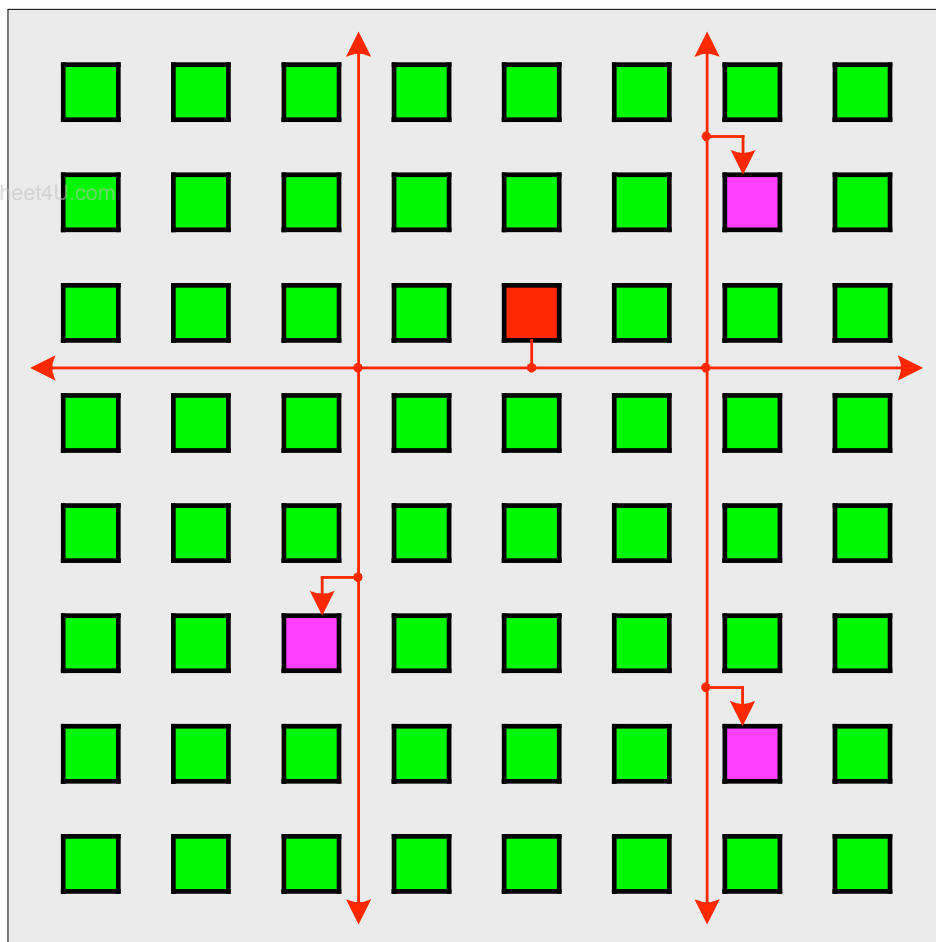


Intra-Segment Interconnect Concept



Intra-Segment Routing Example

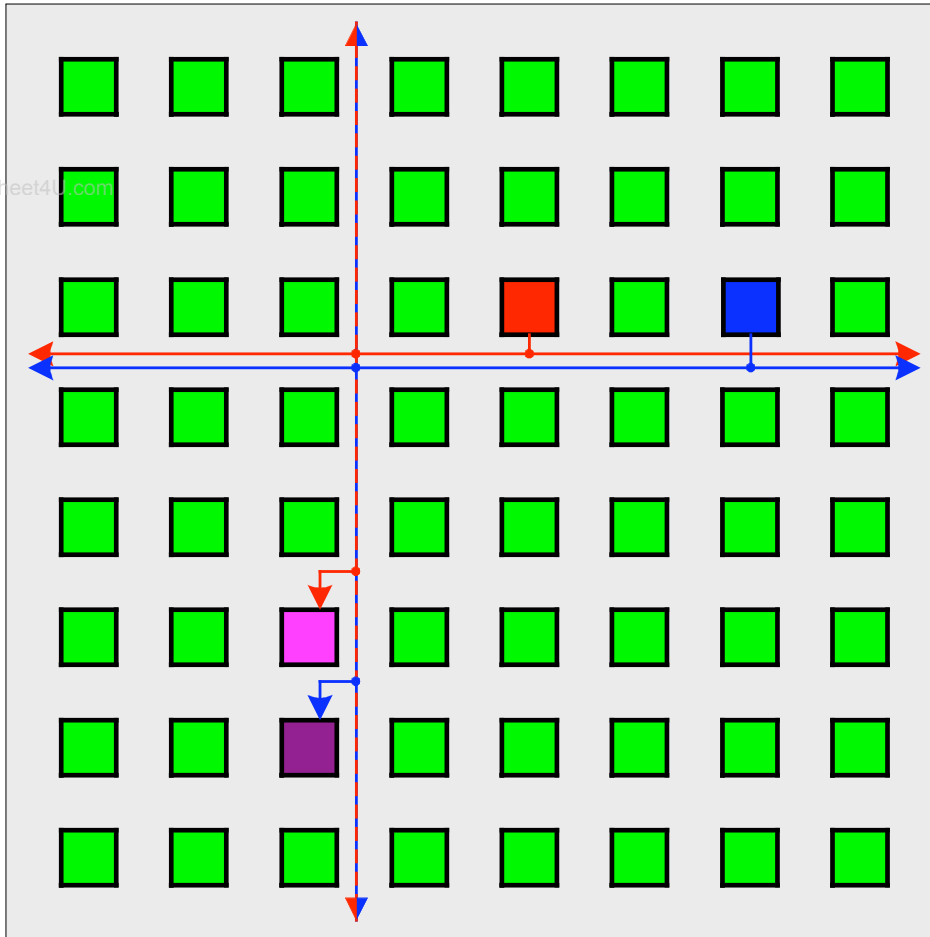
Single net



- **Considering single net**
 - 100% routable from arbitrary source to arbitrary destinations
- **Considering multiple nets (real application)**
 - Contentions may occur
 - Need careful placement to avoid contentions

Intra-Segment Routing Contention

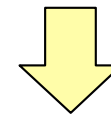
Two nets



V-BUS Contention

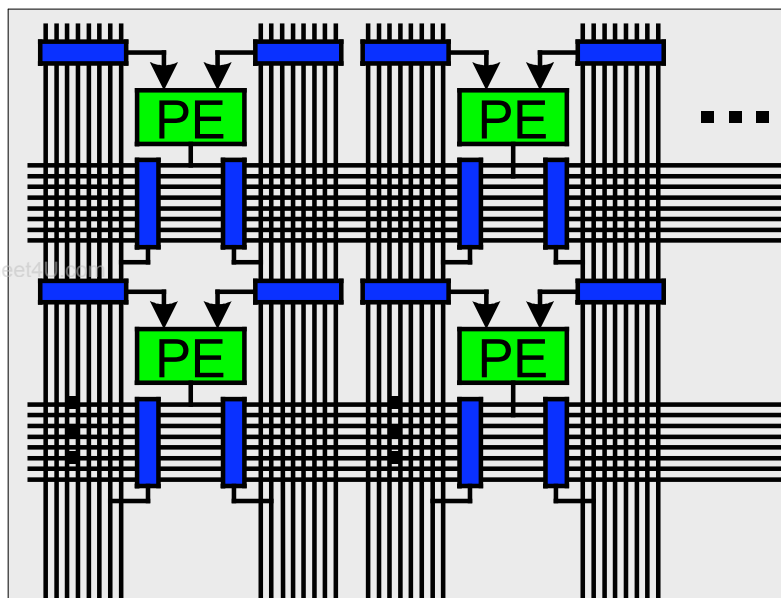
Unroutable if

net1.src.row == net2.src.row
&&
net1.dst.col == net2.dst.col

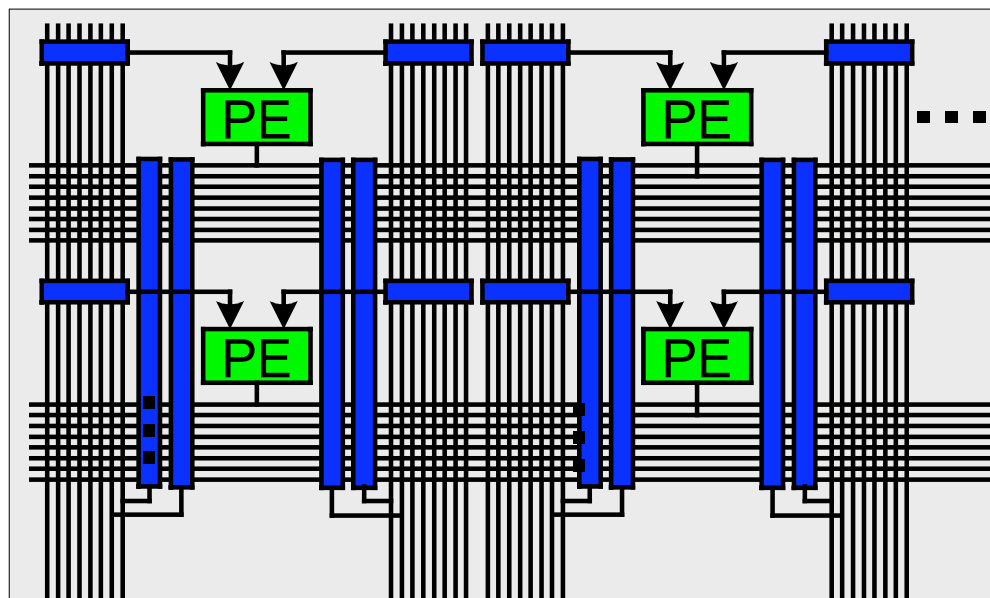
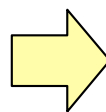


Too restrictive

Revised Intra-Segment Interconnect



“At most one signal
from a row”
to each column-L
(or column-R)



“At most two signals
from two rows”
to each column-L
(or column-R)



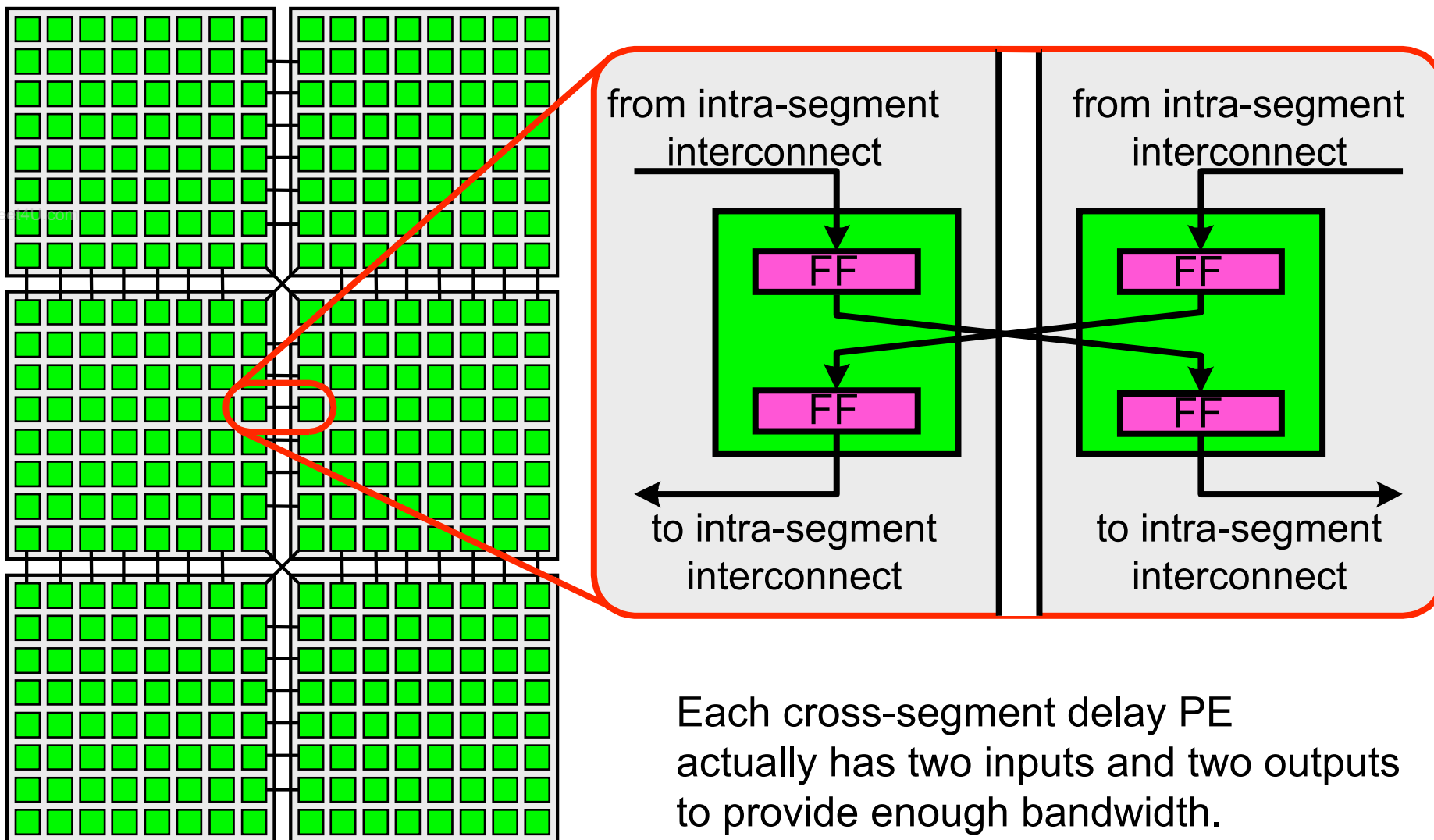
Intra-Segment Interconnect Comparison

Type	MUX structure	MUX Qty. (*1)	Routing delay (*2)
Ideal full-crossbar	64-to-1 x 128 (PE-to-PE)	8064	7 + 6
Rev.1	8-to-1 x 128 (H-to-V) 8-to-1 x 128 (V-to-PE)	1792	7 + 6
Rev.2	16-to-1 x 128 (H-to-V) 8-to-1 x 128 (V-to-PE)	2816	8 + 7

(*1) number of underlying 2-to-1 32-bit MUXs

(*2) levels of FO2 buffers + 2-to-1 MUXs

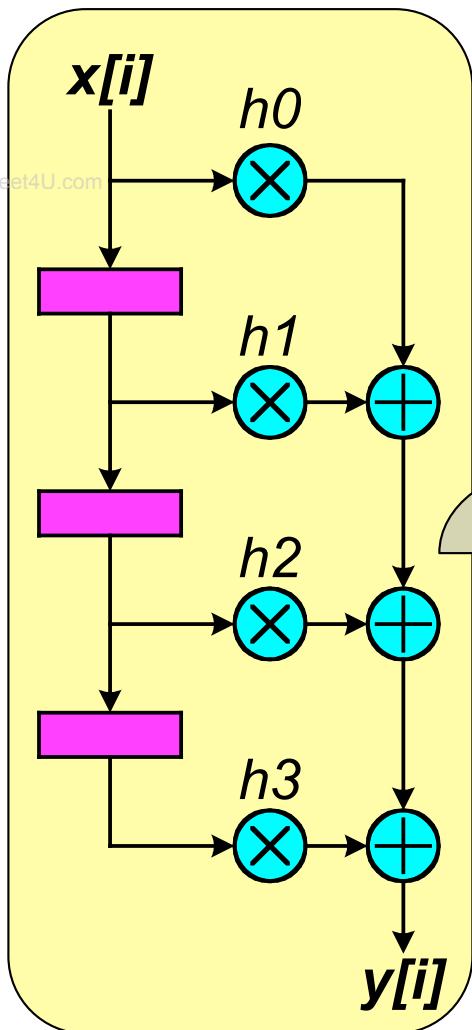
Inter-Segment Routing



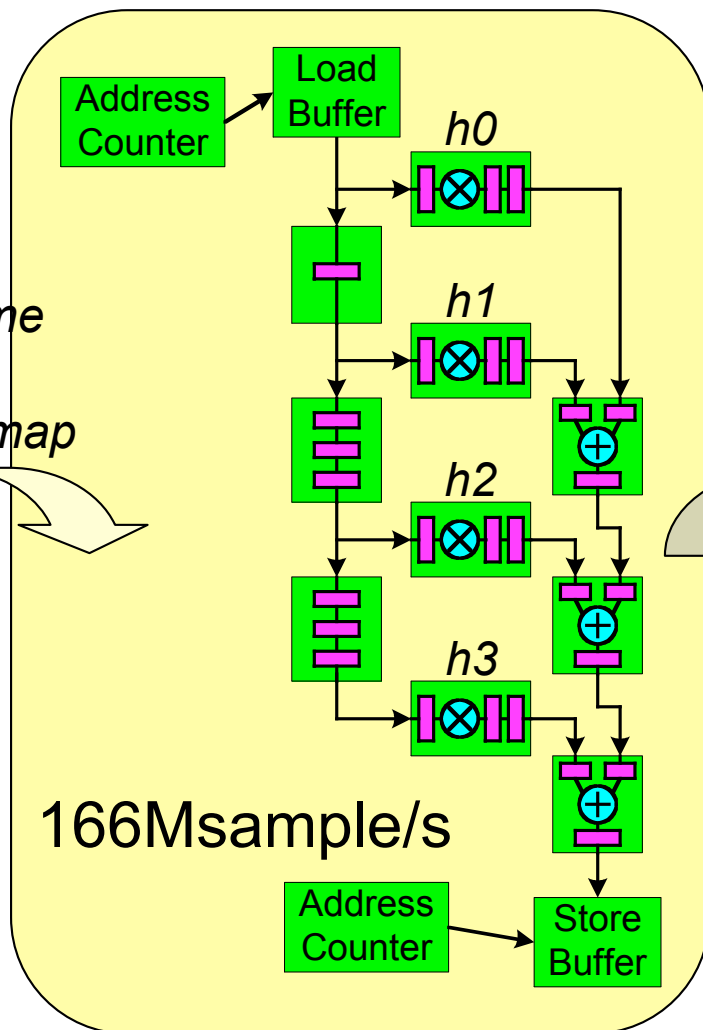
Each cross-segment delay PE actually has two inputs and two outputs to provide enough bandwidth.

Application Example: 4-Tap FIR Filter

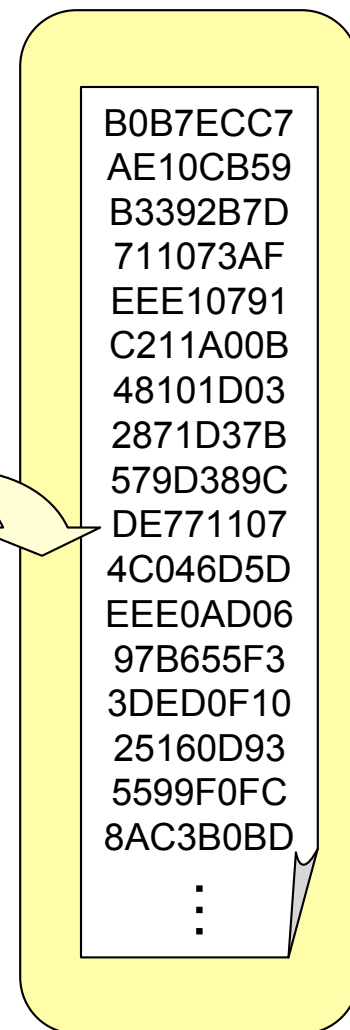
Schematic



PE NetList



Config Data



Retime & Tech-map

Place & Route



Alternative: C-Based Design

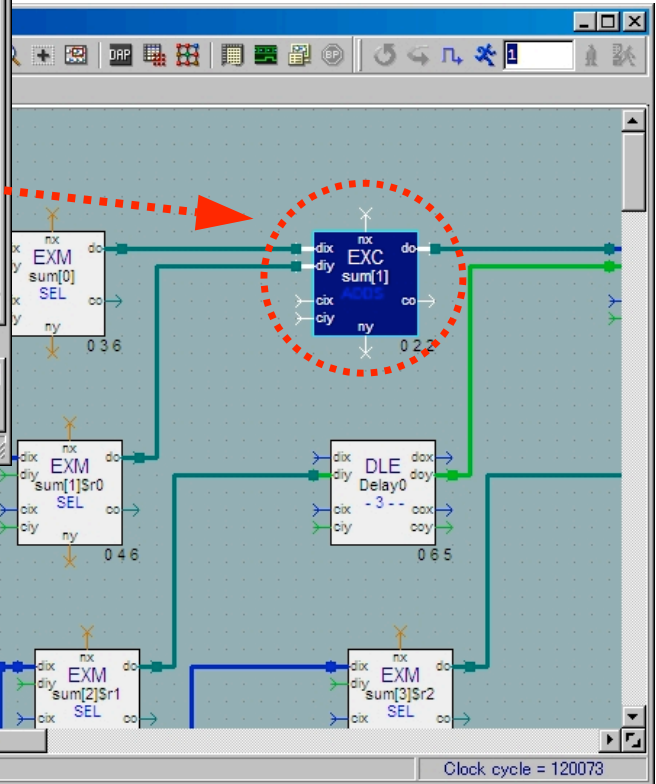
```
int i; /* output data stream */
int nr /* Number of output sample */

int i;
int sum[taps];
delayline int del;

for (i=0; i<nr; i++) {
    del = x[i];

    seq (j=0; j<taps; j++) {
        ifselect (j==0) {
            sum[j] = del[j] * h[j];
        }
        else {
            sum[j] = sum[i-1] + del[j] * h[j];
        }
    }
    r[i] = sum[taps-1];
}
```

Source Code Editor/Debugger



NetList Viewer

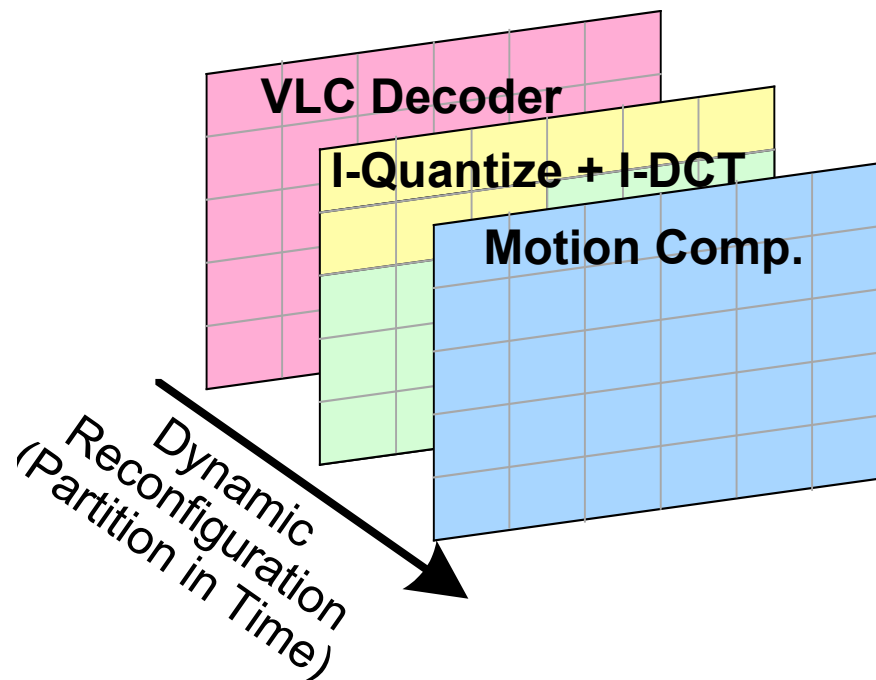
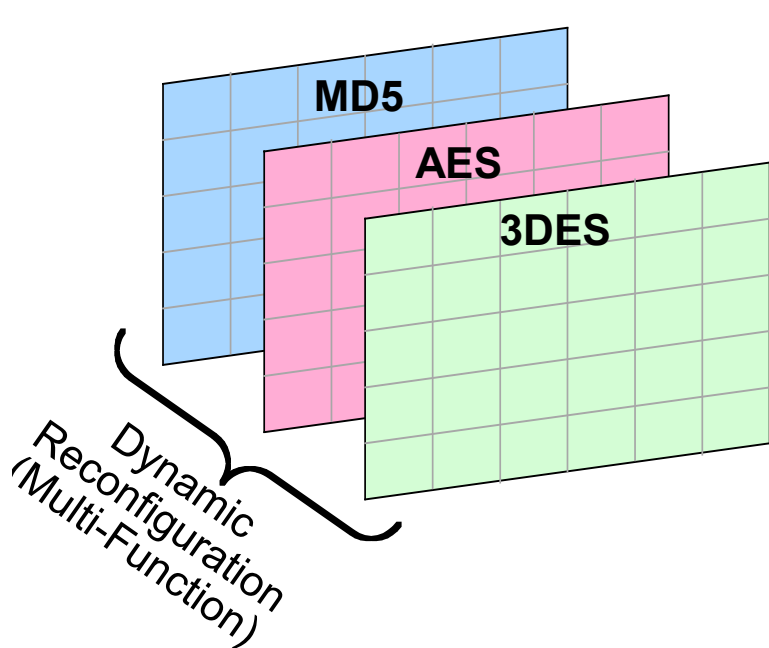


Performance Comparison

Application	Pentium4 3 GHz	DAPDNA-2 166 MHz	Performance Multiple
1024pt FFT (k iterations/s)	8.2	325	40
3x3 Average Filter (M pixels/s)	15.5	666	43
7x7 FIR Filter (M pixels/s)	3.0	166	55
Binary Pixel Expansion / Contraction (M pixels/s)	32.7	666	20
Floyd-Steinberg Error Diffusion (M pixels/s)	11	304	28
Jarvis-Judice-Ninke Error Diffusion (M pixels/s)	6.2	110	18

*Simulation data

- Exploits “temporal computing”
- 4 banks of configuration memory
(1 foreground bank + 3 background banks)
- Can copy from a background bank to the foreground bank (6 kB) in one cycle





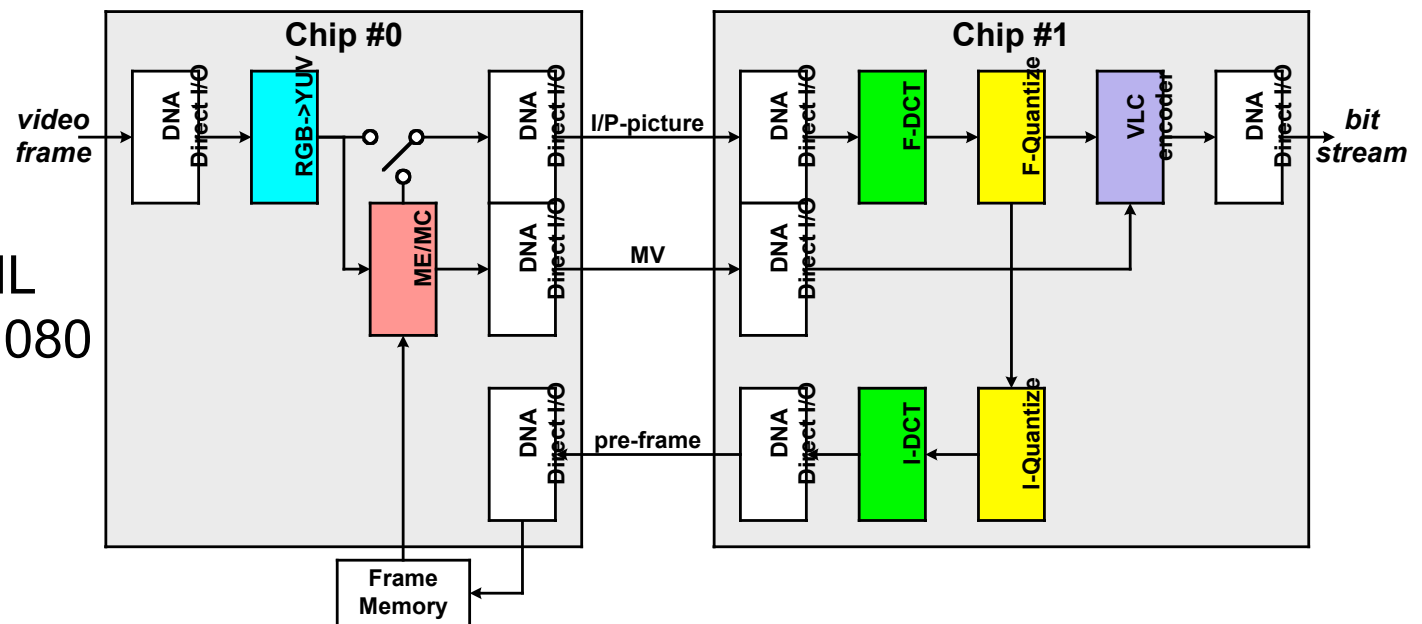
Advanced Usage: Multiple Chips

- Exploits “spatial computing”
- Six 32-bit I/O channels for inter-chip interconnect
- I/O channels can be treated as delay elements

www.DataSheet4U.com

Example: MPEG-2 realtime encoder using 2 chips

Format : MP@HL
Resolution : 1920x1080
Frame rate: 30 fps
Latency : 46 ms





DAPDNA-2 Summary

- **High performance**
 - thanks to massively parallel processing elements
- **Easy to use**
 - thanks to fixed-frequency coarse-grained ALUs
- **Dynamically reconfigurable in one cycle**
- **Scalable via high-bandwidth I/O channels**
- **Schematic-based or C-based application design flow**



<http://www.ipflex.com/>