Novel Multimedia Instruction Capabilities in VLIW Media Processors

J. T. J. van Eijndhoven [1,2]

F. W. Sijstermans [1]

(1) Philips Research Eindhoven

(2) Eindhoven University of Technology

The Netherlands

eijndhvn@natlab.research.philips.com

Contents

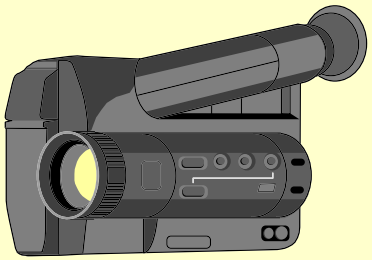
- Background
- Towards a new architecture
- Starting point
- Approach
- New features
- Example
- Conclusion

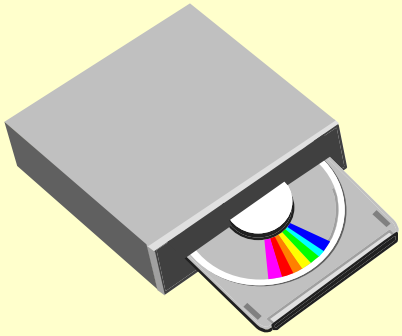
Background

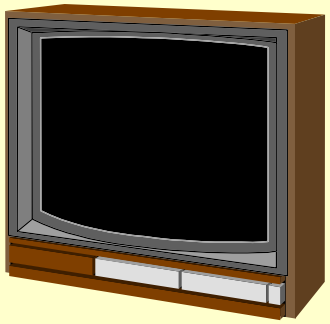
- Philips Semiconductors has a TriMedia product line
- Featuring a VLIW processor core and on-chip peripherals
- Intended for Audio/Video media processing
- In consumer electronic devices

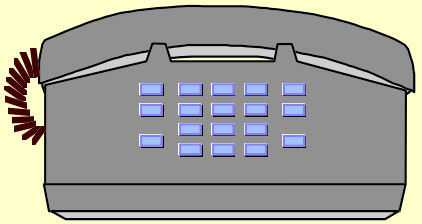
A next-generation VLIW core architecture was developed at Philips Research

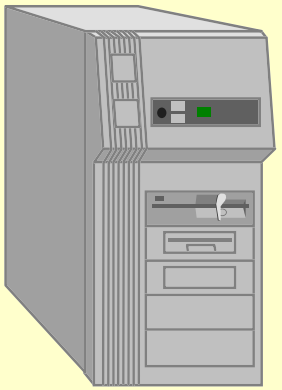
TM1000 overview

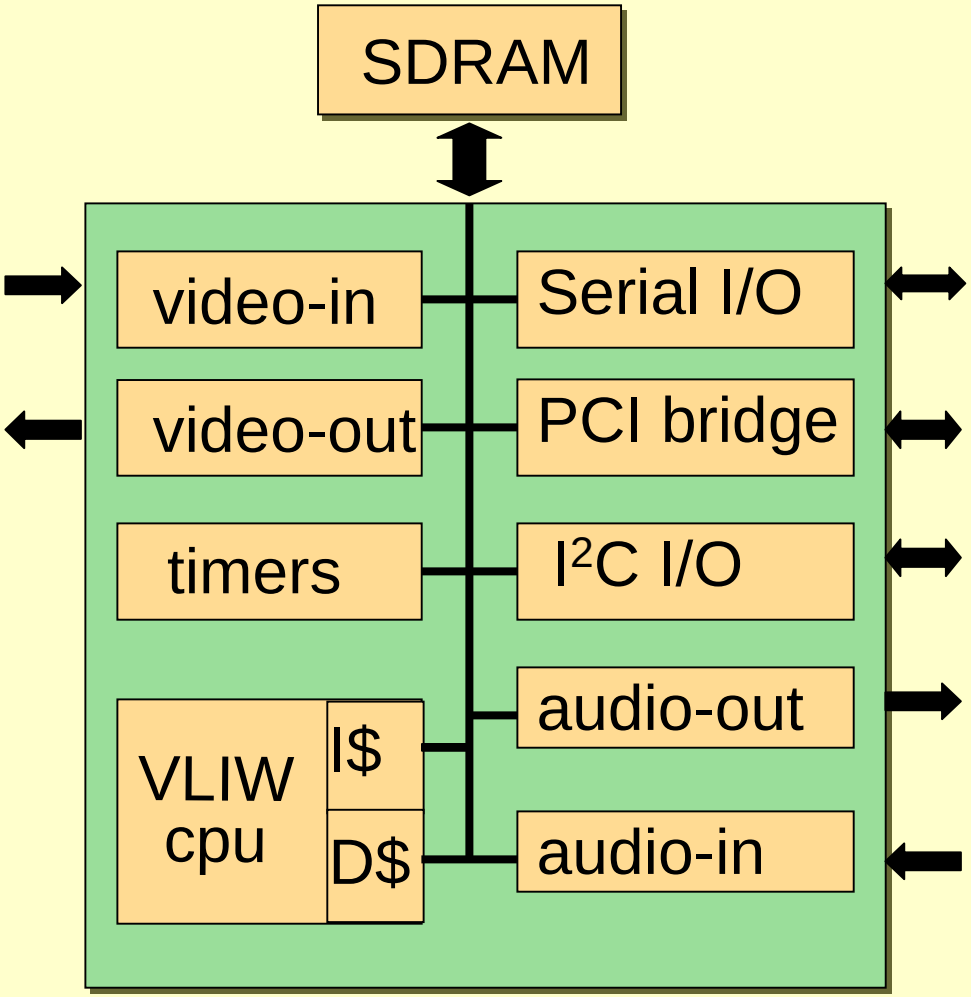




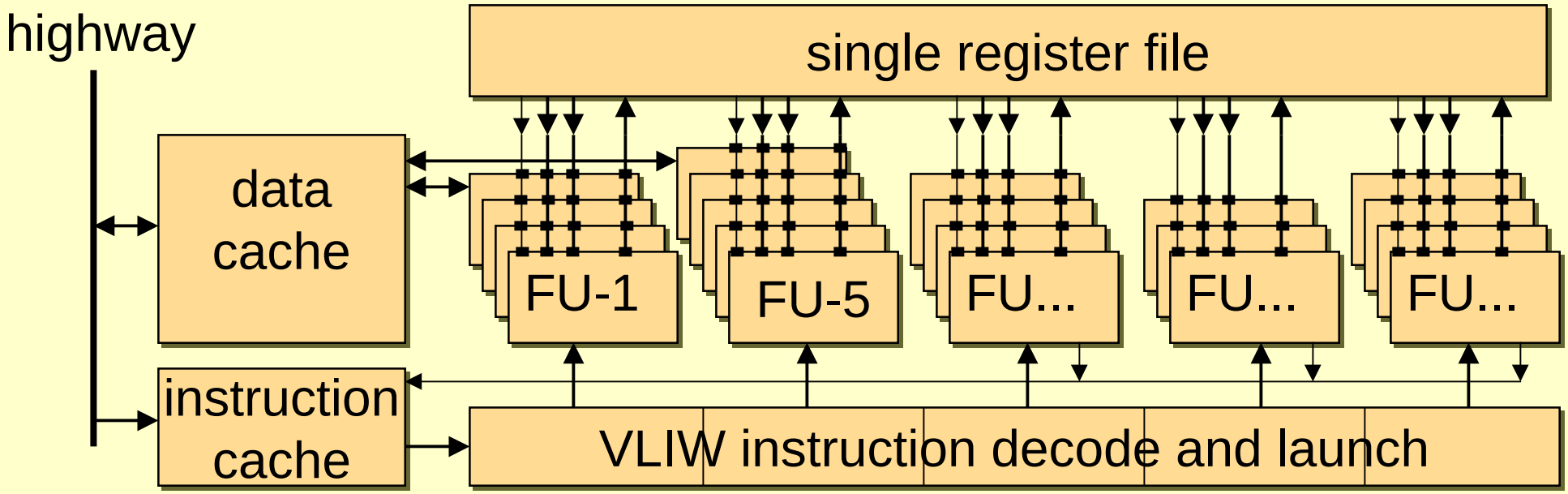








TM1000 VLIW core



32 KB instr cache
16 KB data cache,
quasi dual ported,
8-way set associative

128 words x 32 bits register file

5 ALU,	5 const,	2 shift,	3 branch
2 I/FPmul,	2 FPalu,	1 FPdivsqrt,	1 FPcomp
2 loadstore,	2 DSPalu,	2 DSPmul	

Pipelined, latency 1 to 3 cycles (except FPdivsqrt)

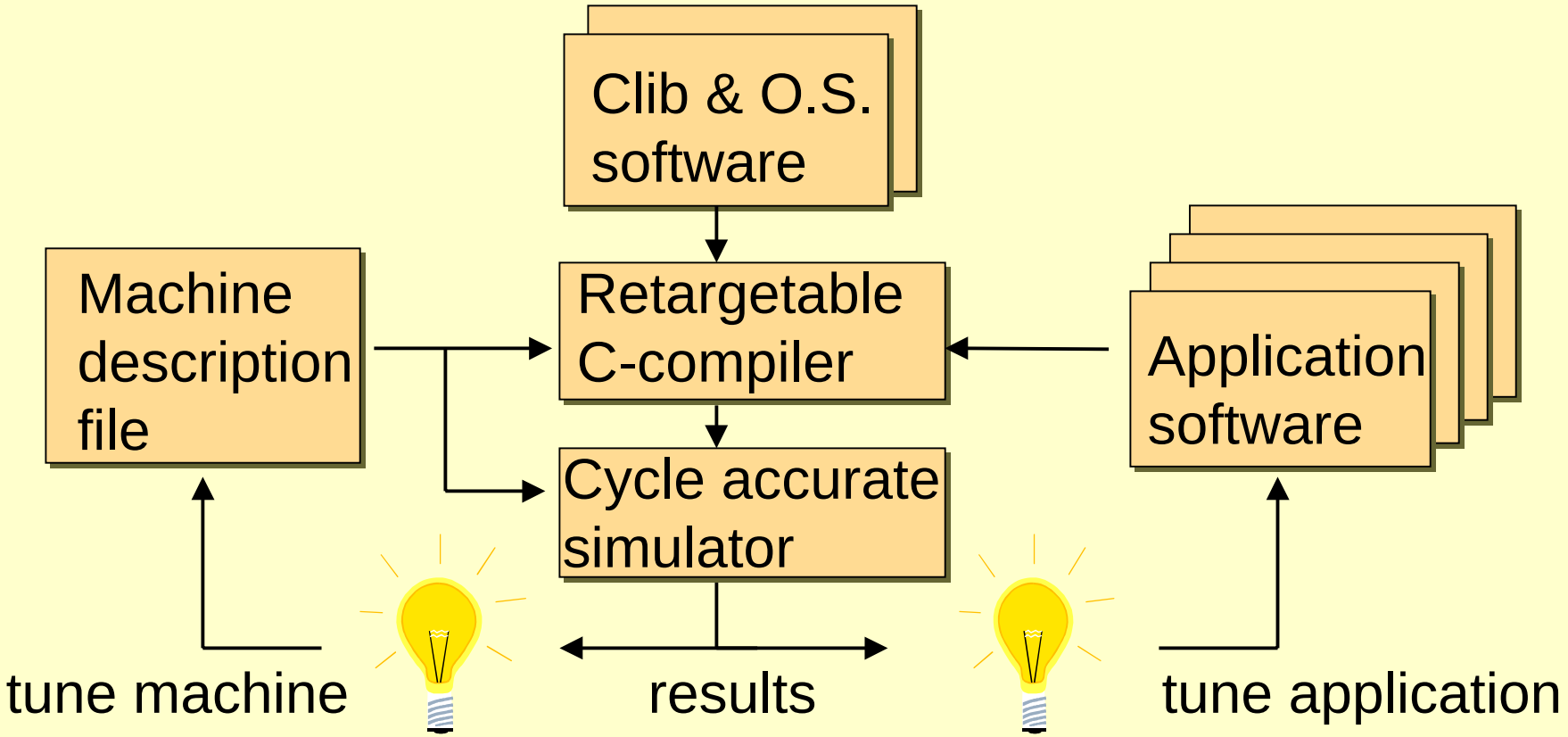
Next generation architecture

Significantly improve VLIW processor performance by:

- Richer instruction set
- Wider data words
- Improved cache behavior
- Higher clock frequency

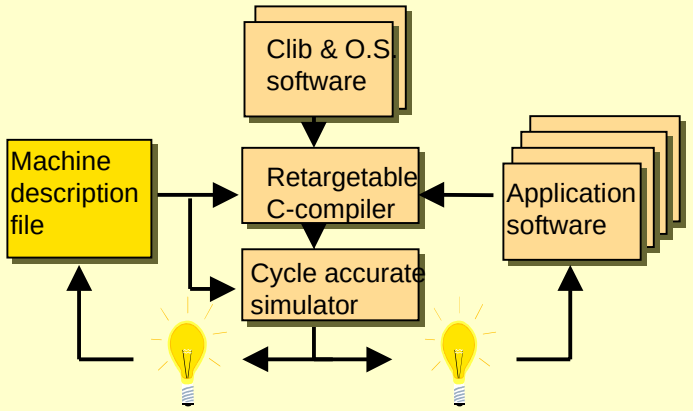
Approach

Quantitative design space exploration:



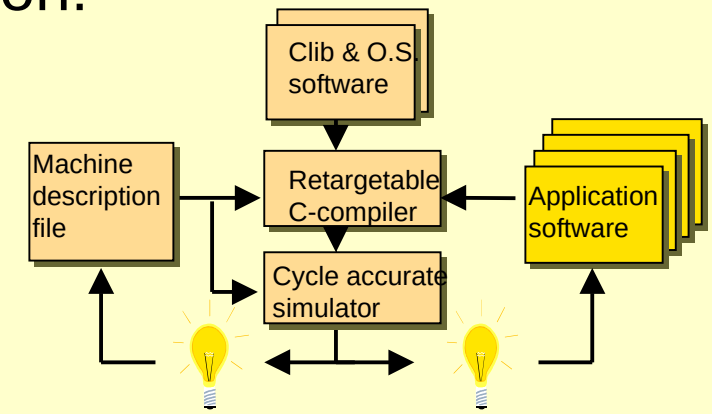
Machine description

```
CPU
     ISSUESLOTS 5
     FUNCTIONAL UNITS
          alu  SLOT 1 2 3 4 5 LATENCY 1
               OPERATIONS
                    iadd(12), isub(13),
                    igtr(15), igeq(14),
          dspalu
               SLOT 1 3       LATENCY 2
               OPERATIONS
                    dspiadd(66), dspuadd(67)
     REGISTERS  r SIZE 32 NUMBER 128;
     READ BUSES REGISTERS r NUMBER 10;
OPERATIONS
     SIGNATURE (r:r,r->r) PURE iadd, isub,
     SIGNATURE (r:PAR,r->r) PARAMETER (0 to 127) PURE iaddi,
     SIGNATURE (r:r,r->r) LOADCLASS ld32x,
```



Application Software

Applications used for design space exploration:
- MPEG2 decode, in particular IDCT
- Television progressive scan conversion:
  natural motion estimation & compensation
- 3D graphics library
- AC3 digital audio



Source code optimization towards architecture:
- analyse computation in critical sections: choice of algorithm
- vectorization of data and loops
- insertion of ‘multimedia’ machine operations
- provide compiler hints (*restrict* pointers, loop unrolling)

Obtain recommendations for new ‘multimedia’ operations!

New Architecture

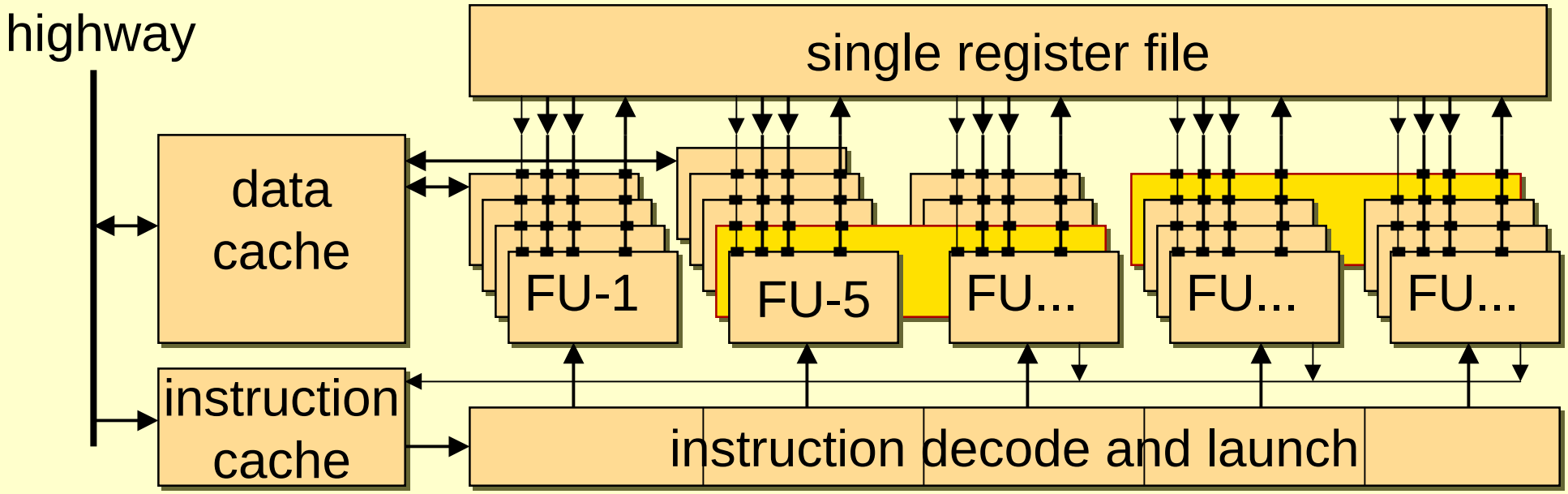
- Single registerfile of 128 words x 64 bits
- Maintain 5 issue slots
- Treat 64-bit words as vectors of 8-, 16-, or 32-bit data elements,
- Provide an extensive set of operations to support these vectors, as signed or unsiged data, clipped or wrap-around arithmetic.
- Provide a limited set of special operations to speed up particular applications

Introduction of a new capability: SuperOperations

SuperOperations

- A (2-slot) SuperOp can accommodate:
  - 4 argument registers
  - 2 result registers
- Its functional unit can thus implement a powerful operation.
- The SuperOp occupies 2 adjacent slots in the VLIW instruction format.
  - Fitting the basic instruction format: fixed fields for registers.
  - Fitting the available ports to the register file.
- Can be supported in the architecture with very little overhead.

SuperOperations in Hardware

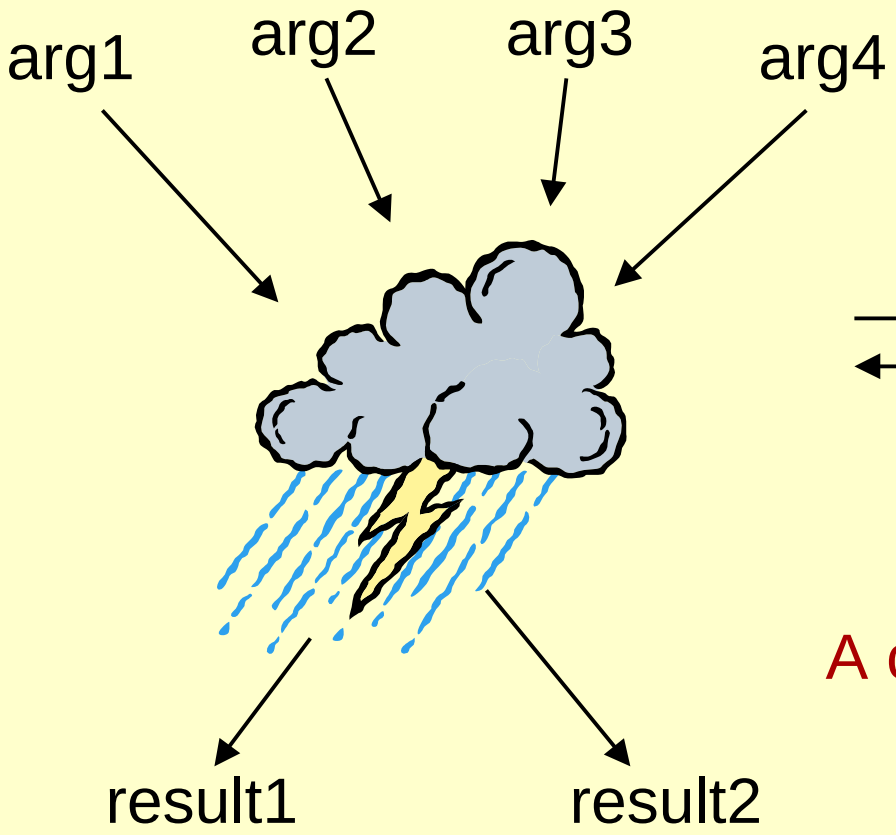


- adjacent instruction slots
- regular decode (location of fields)
- existing register file ports

SuperOperations in Software

- SuperOps are available in C programs as procedure calls.
  (as all other multimedia and SIMD operations)
- The C compiler maps these to a single machine operation.
  (for dual-output this requires optimizing away the & operator)
- The instruction scheduler is aware of the (multi-) slot restrictions:
  - Slot assignment becomes more complex.
    (feasible shuffles of operations in a single instruction)
  - Register allocation requires some adjustment.

SuperOperation definition

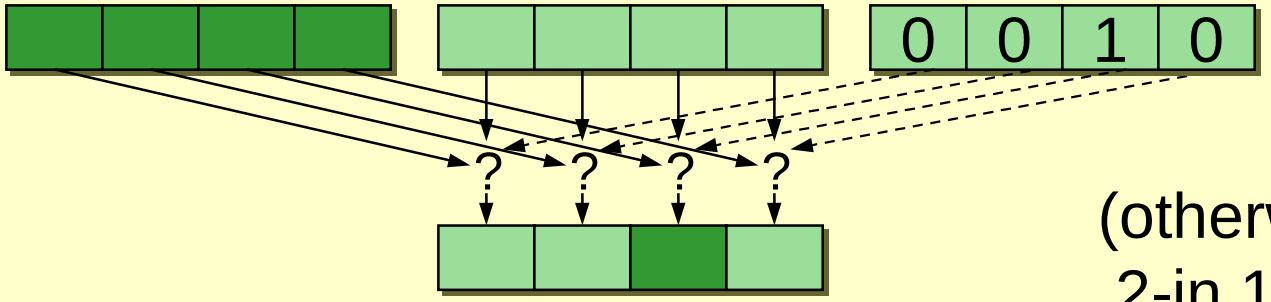


Multimedia Software:
- MPEG
- Television
- 3D graphics
- audio

A complex design space optimization!

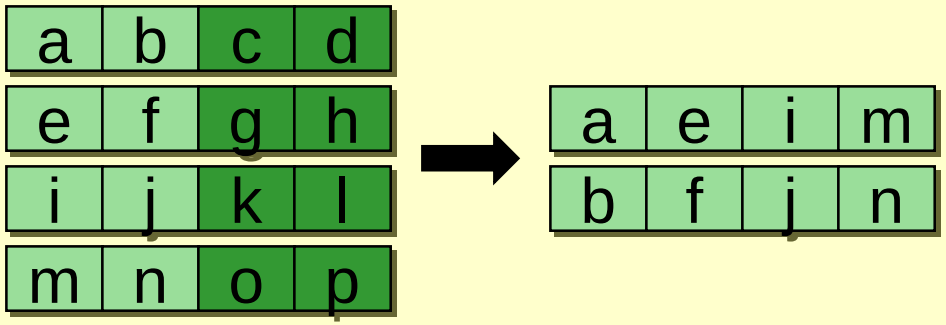
SuperOp examples (1)

vector multiplex: 1 result vector, 2 argument data vectors,
a 3rd argument specifying a choice for each 16-bit element.



(otherwise 3 simple
2-in 1-out operations)

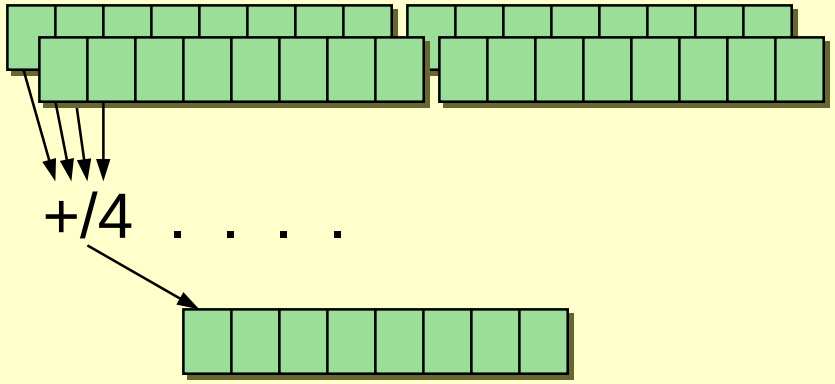
Transpose half-word high (and -low):
4 data argument vectors of 16-bit elements, 2 result vectors



(otherwise 6)

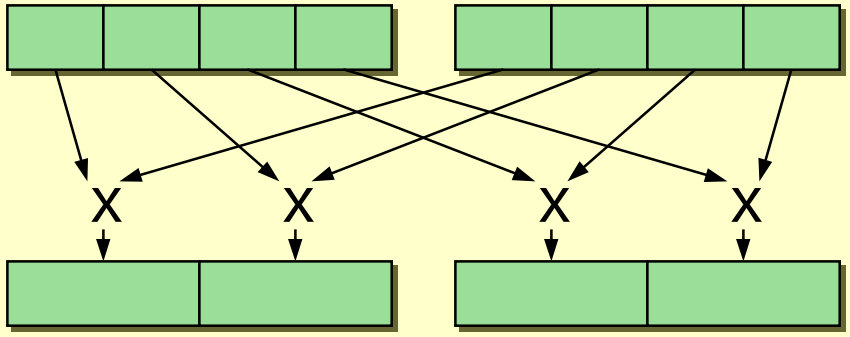
SuperOp examples (2)

2-dimensional half-pixel average:



(otherwise 15)

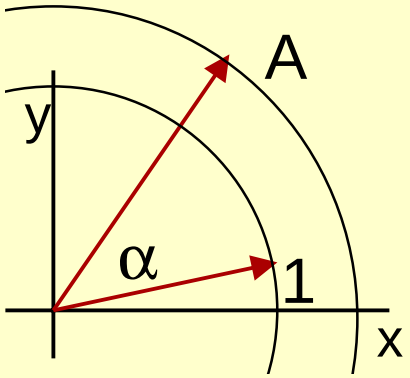
Multiply to double precision:



(otherwise 2)

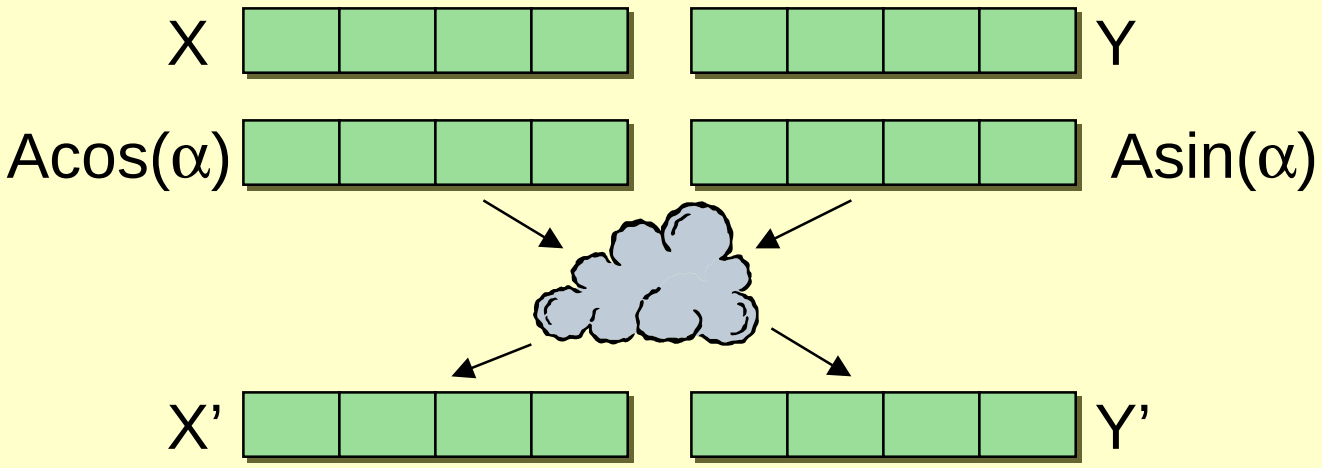
SuperOp examples (3)

Rotate:



$X' = A\cos(\alpha)\, X + A\sin(\alpha)\, Y$

$Y' = A\cos(\alpha)\, Y - A\sin(\alpha)\, X$



(otherwise 6)

Motion Compensation with SuperOps

Motion compensation from MPEG2, block of 16x8 pixels, with half-pixel accuracy (including loads and stores):

MotionComp [cycles]	No 2D-average superop	Have 2D-average superop
No funshift superop	82	39
Have funshift	70	31

The IDCT example

The IDCT is an important computational kernel in MPEG.
The 2-dimensional 8x8 point IDCT was implemented in C, and compiled and simulated with the created tools.
It operates entirely on (vectors of) 16-bit data elements.
The generated code includes:

- The standard function-call stack mechanism.
- Initial load operations to get the data into the register file.
- Final write operations to store back the result.
- Immediates for multiplication constants.

Simulation on the target machine showed IEEE 1180 accuracy compliancy.

IDCT with SuperOps

2D-IDCT [cycles]	No rotate superop	Have rotate superop
Transpose without superop	60	47
Have transpose superop	56	43
No transpose, have special IDCT ops	46	37

The IDCT result

The current architecture reaches **56 cycles** (5-slot VLIW, 64 bit)

This is to be compared with:

- 201 cycles for the NEC V830R/AV[1] (2-way SS, 64-bit, 200MHz)
- 247 cycles for the TI TMS320C62[2] (8-slot VLIW, 32-bit, 200MHz)
- 500 cycles for the Mitsubishi D30V[3] (2-way SS, 32-bit, 200MHz)
- 147 for the HP PA-8000 with MAX-2[4] (2-way SS, 64-bit, 240MHz)
- 160 cycles for the TM-1000 (5-slot VLIW, 32-bit, 100MHz)
- [500 for Pentium II with MMX, including dequantization stage[5]]

(But these are available now)

1] K. Suzuki, T. Arai, et.al., *V830R/AV: Embedded Multimedia Superscalar RISC processor*, IEEE Micro, March 1998, pp. 36-47
2] N. Seshan, *High VelociTI Processing*, IEEE Signal Processing Magazine, March 1998, pp.. 86-101
3] E. Holmann, T. Yoshida, et.al., *Single Chip Dual-Issue RISC Processor for Real-Time MPEG-2 Software Decoding*, J. VLSI Signal proc., 18, 1998, 155-165
4] R. Lee, *Effectiveness of the MAX-2 Multimedia Extensions for PA-RISC 2.0 processors*, HotChips IX symposium, Aug. 1997, pp. 135-148
5] Intel, *Pentium II Application note 886*, 1997, http://developer/intel/com/drg/pentiumII/appnotes//886.htm

Conclusion

- An architecture has been defined for a new generation multimedia processor in the TriMedia product line.
  It was recently transferred to Philips Semiconductors for physical design. (More details are announced at Microprocessor Forum ‘98)
- SuperOperations, occupying multiple adjacent slots in the VLIW instruction, are added as new concept. For specific occasions, they allow considerable speedup with limited architectural consequences.
- A retargetable C-compiler, instruction-scheduler and simulator are used to tune the architecture and quantify application results.