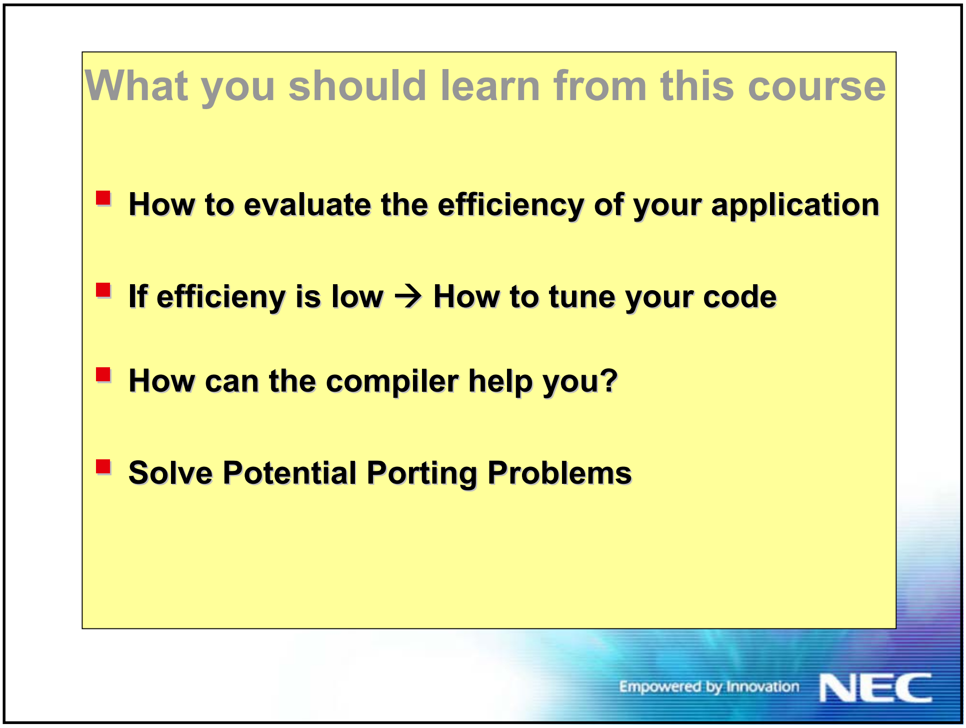
Empowered by Innovation

NEC

NEC SX-6
User Training HLRS
16.3.2004

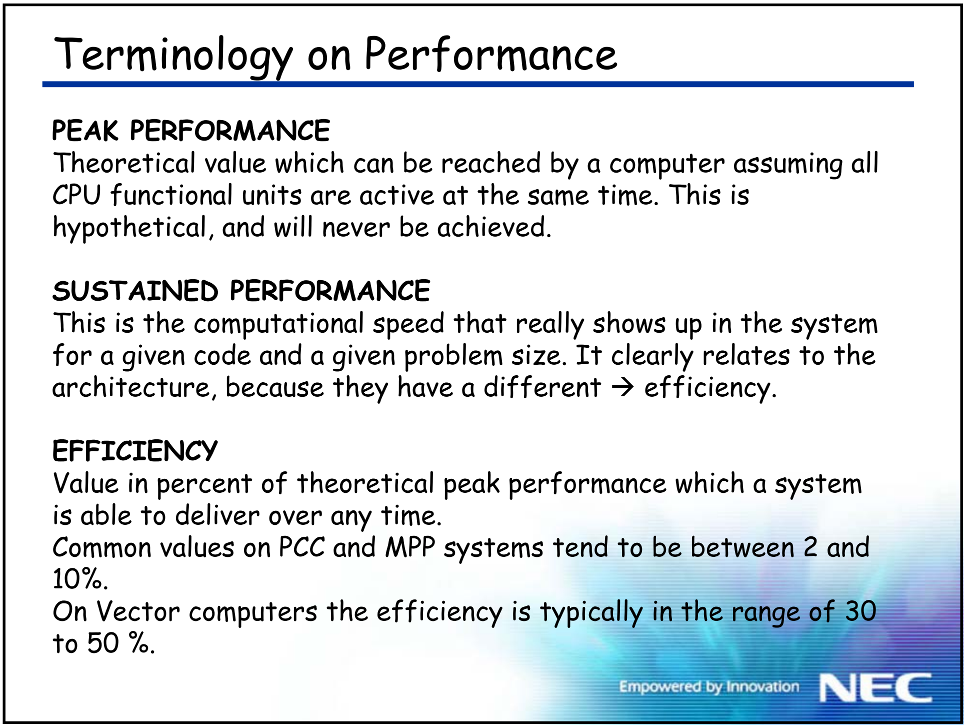
Dr. Th. Schoenemeyer, NEC HPCE
Application Specialist
tschoenemeyer@hpce.nec.com



What you should learn from this course

- How to evaluate the efficiency of your application
- If efficieny is low → How to tune your code
- How can the compiler help you?
- Solve Potential Porting Problems

Empowered by Innovation NEC



Terminology on Performance

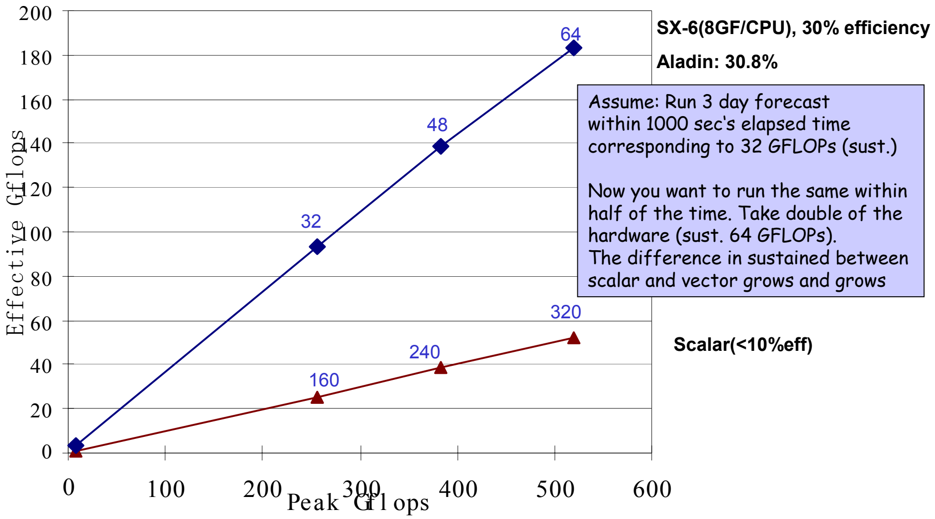
PEAK PERFORMANCE
Theoretical value which can be reached by a computer assuming all CPU functional units are active at the same time. This is hypothetical, and will never be achieved.

SUSTAINED PERFORMANCE
This is the computational speed that really shows up in the system for a given code and a given problem size. It clearly relates to the architecture, because they have a different → efficiency.

EFFICIENCY
Value in percent of theoretical peak performance which a system is able to deliver over any time.
Common values on PCC and MPP systems tend to be between 2 and 10%.
On Vector computers the efficiency is typically in the range of 30 to 50 %.

Empowered by Innovation NEC

Typical Weather Code



SX-6(8GF/CPU), 30% efficiency

Aladin: 30.8%

Assume: Run 3 day forecast within 1000 sec's elapsed time corresponding to 32 GFLOPs (sust.)

Now you want to run the same within half of the time. Take double of the hardware (sust. 64 GFLOPs).
The difference in sustained between scalar and vector grows and grows

Scalar(<10%eff)

Empowered by Innovation NEC

What you should learn from this course

- How to evaluate the efficiency of your application

Empowered by Innovation NEC

It's all about this table

```
    ******  Program Information  ******
  Real Time (sec)          :        142.373845
  User Time (sec)          :        122.693250
  Sys  Time (sec)          :          1.331073
  Vector Time (sec)        :         65.895359
 Inst. Count               :      19621582641.
 V. Inst. Count            :       2791820381.
 V. Element Count          :     216956074637.
 FLOP Count                :      97760445434.
  MOPS                     :       1905.449870
  MFLOPS                   :        796.787481
  VLEN                     :         77.711330
  V. Op. Ratio (%)         :         92.801205
  Memory Size (MB)         :        496.031250
  MIPS                     :        159.923897
  I-Cache (sec)            :          1.852328
  O-Cache (sec)            :         15.546471
  Bank (sec)               :          3.069632

  Start Time (date)   :  2002/06/24 11:32:19
  End   Time (date)   :  2002/06/24 11:34:42
```

- How to create
- Understand
- Improve

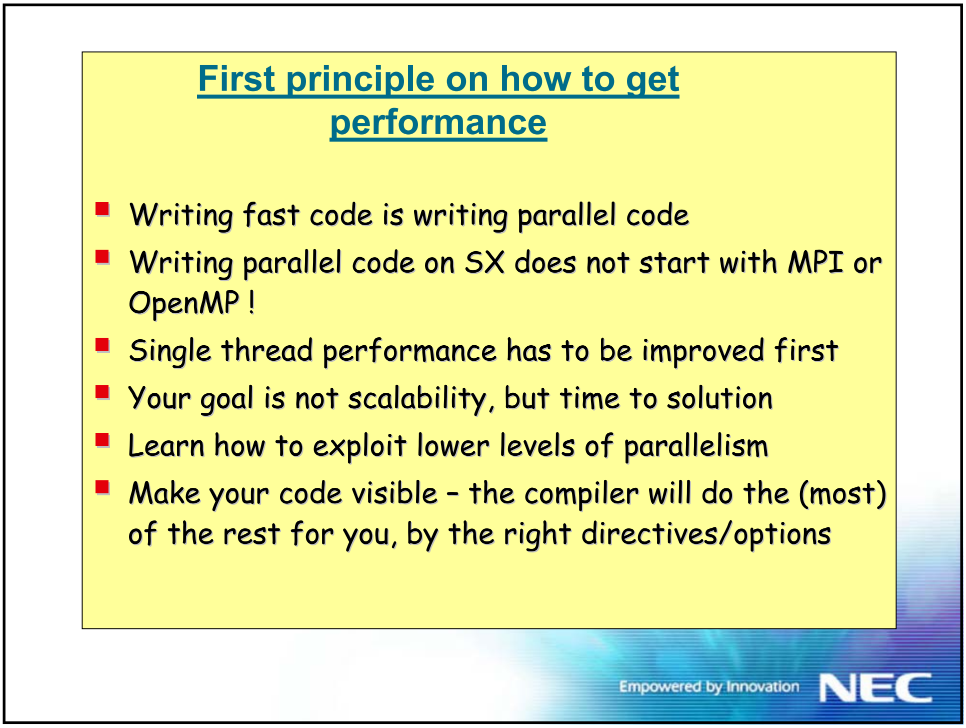
Use it always, It's like checking the gasoline consumption of your car …

Empowered by Innovation NEC

What you should learn from this course

- If your ported code crashes → How to solve
- If efficieny is low → How to tune your code

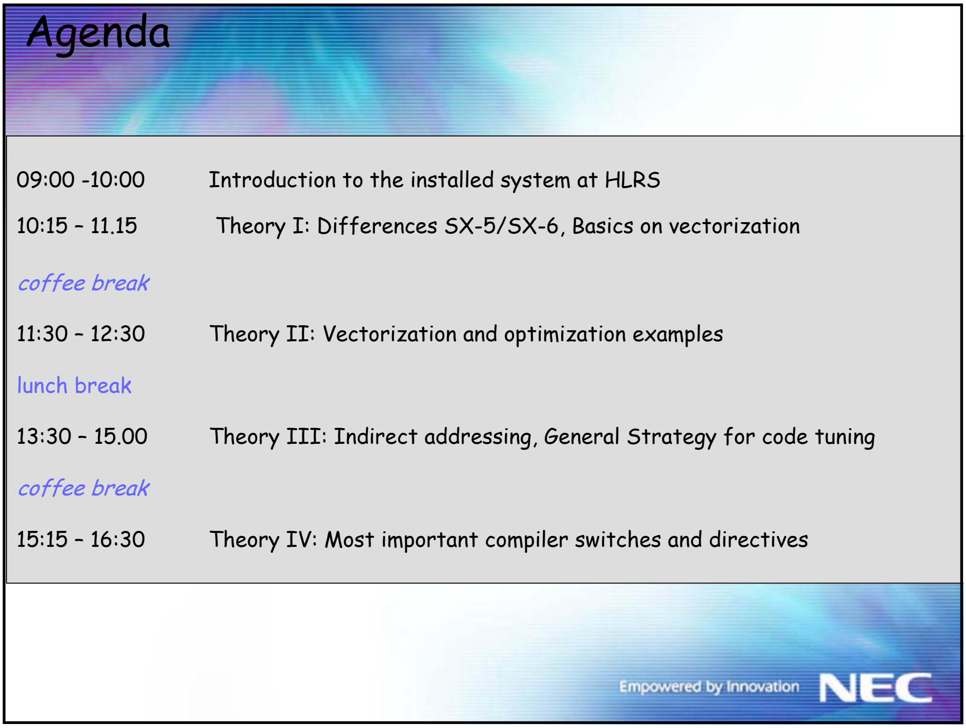
Empowered by Innovation NEC



First principle on how to get performance

- Writing fast code is writing parallel code
- Writing parallel code on SX does not start with MPI or OpenMP !
- Single thread performance has to be improved first
- Your goal is not scalability, but time to solution
- Learn how to exploit lower levels of parallelism
- Make your code visible – the compiler will do the (most) of the rest for you, by the right directives/options

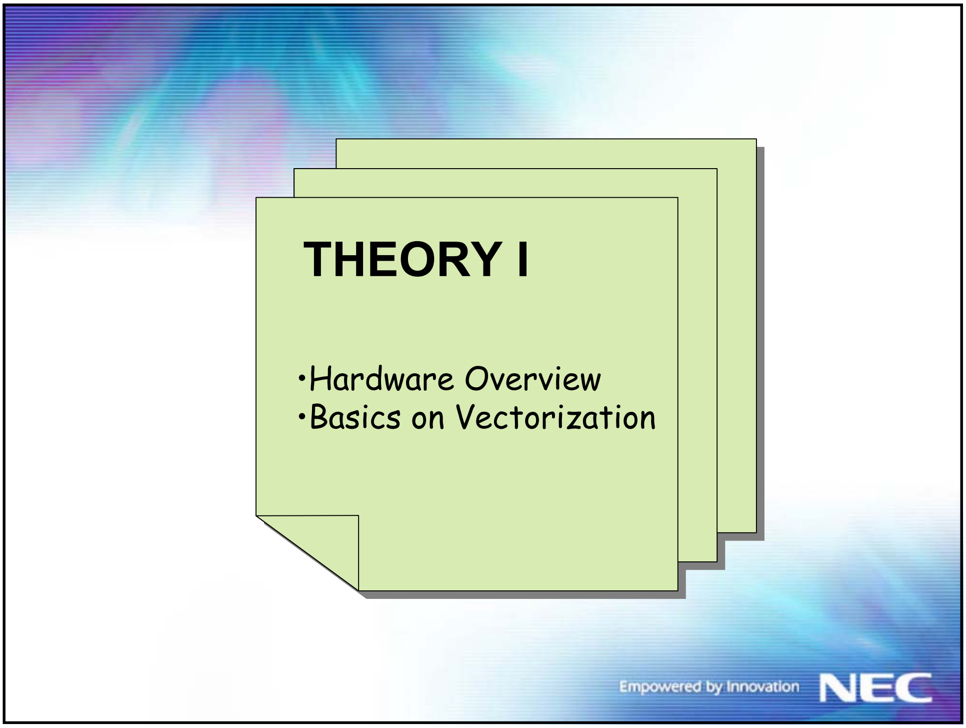
Empowered by Innovation NEC



Agenda

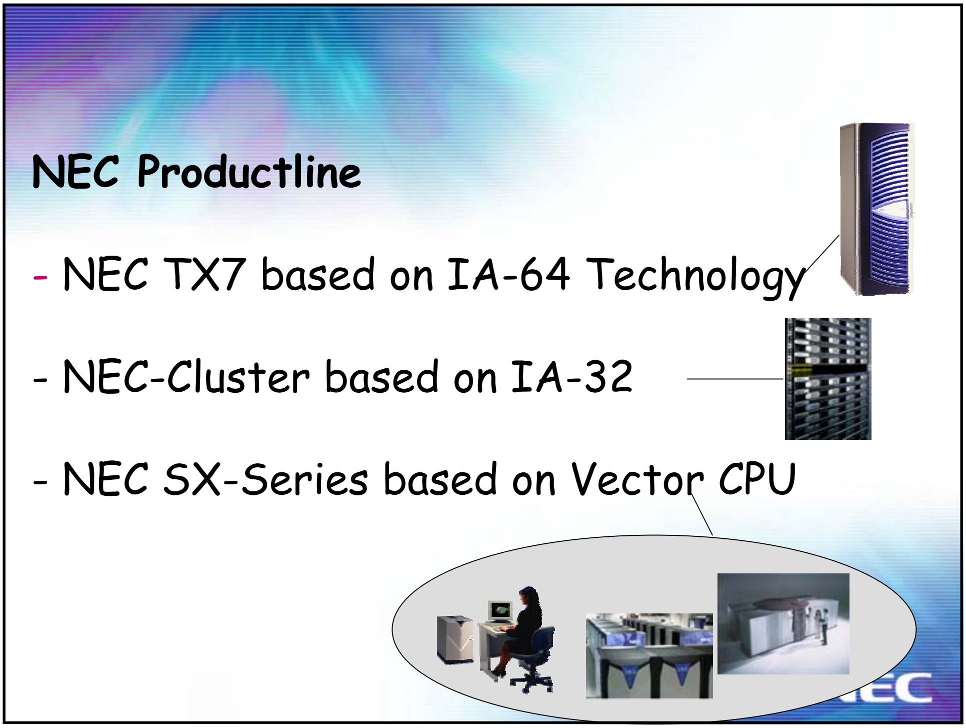
09:00 -10:00	Introduction to the installed system at HLRS
10:15 – 11.15	Theory I: Differences SX-5/SX-6, Basics on vectorization
coffee break	
11:30 – 12:30	Theory II: Vectorization and optimization examples
lunch break	
13:30 – 15.00	Theory III: Indirect addressing, General Strategy for code tuning
coffee break	
15:15 – 16:30	Theory IV: Most important compiler switches and directives

Empowered by Innovation NEC



THEORY I

- Hardware Overview
- Basics on Vectorization



NEC Productline

- NEC TX7 based on IA-64 Technology
- NEC-Cluster based on IA-32
- NEC SX-Series based on Vector CPU

Overview NEC TX-7

The TX-7 Series Server family uses the Intel® Itanium®2 Processor

- up to 1,5 GHz clock speed
- up to 32 CPU's
- up to 256 GB Main Memory
- 6.4 GB/s system bus bandwidth
- 3-6 MB integrated L3 cache
- 64-bit memory addressing
- Peak performance up to 6 GFLOPS per processor
- Linpack: 171.61 GFLOPs with full node (15th July 2003)
- HP-UX, Windows, NEC Linux

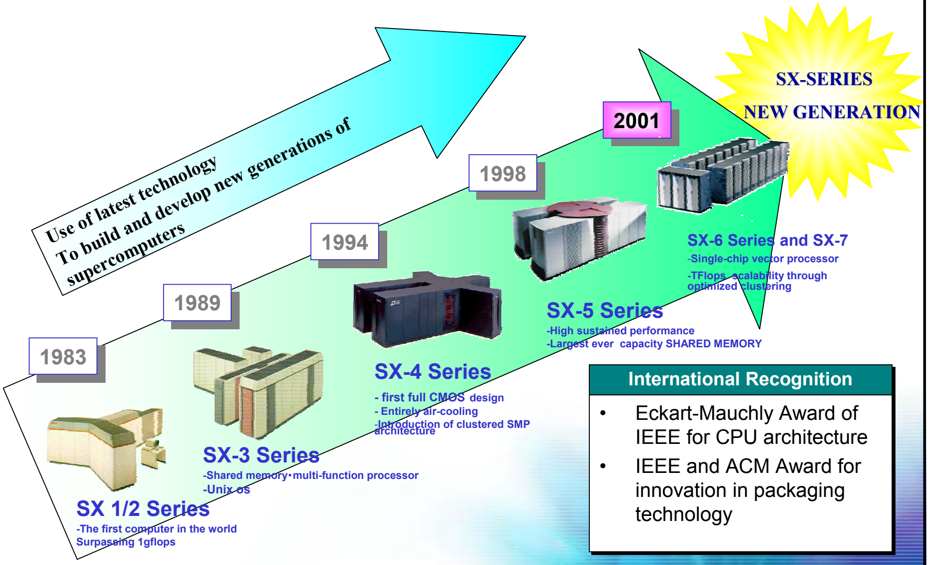
Empowered by Innovation NEC

NEC IA-32 Cluster

- **NEC Cluster solution for crash simulation**
  - 4 fully equipped Racks
  - 256 Xeon CPUs
- **Intel Xeon based Server**
- **Available: Gigabit-Eth., Myrinet, Quadrics, Infiniband**
- **System Support from NEC**

Empowered by Innovation NEC

NEC‘s SX-Series: Innovation since 1983


Use of latest technology
To build and develop new generations of supercomputers
SX-SERIES
NEW GENERATION
2001
1998
1994
1989
1983
SX-6 Series and SX-7
-Single-chip vector processor
-TFlops scalability through optimized clustering
SX-5 Series
-High sustained performance
-Largest ever capacity SHARED MEMORY
SX-4 Series
- first full CMOS design
- Entirely air-cooling
-Introduction of clustered SMP architecture
SX-3 Series
-Shared memory･multi-function processor
-Unix os
SX 1/2 Series
-The first computer in the world Surpassing 1gflops

International Recognition

- Eckart-Mauchly Award of IEEE for CPU architecture
- IEEE and ACM Award for innovation in packaging technology

Empowered by Innovation NEC

NEC’s Goals: The Vector Architecture

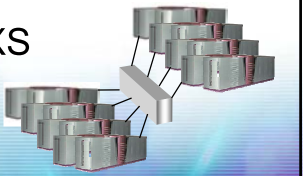
- **Highest single CPU performance**
- **High sustained performance**
- **Extremely high bandwith to Memory**
- **High performance with comparatively small number of CPUs**
- **Same manufacturing technology like conventional CPUs**
- **Provide Software to use it effectively**

Empowered by Innovation NEC

SX-5 The Specifications

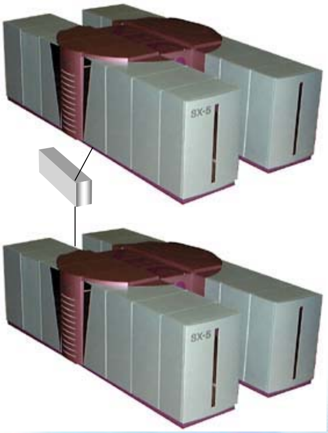
- Single Node
  - Up to 128 GFLOPS
    With 16 x 8 GFLOPS Processors
  - Up to 128 GBytes Shared Main Memory

- Multi Node
  - Up to 4 TFLOPS
  - Up to 32 Nodes Using SX-5 IXS
  - Up to 512 Processors
  - Up to 4 TBytes Main Memory



SX-5 HLRS Installation

- 2 Nodes
  - 128 GFLOPS
    With 32 x 4 GFLOPS
    Processors
  - 80 GBytes Shared
    Main Memory



Our current High-End product SX-6+

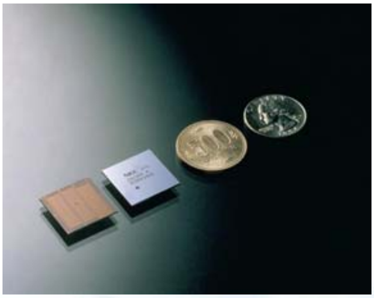
- High Performance Vector Supercomputer
- 9 GF / CPU
- 72 GF/ Node
- Maximum 128 GB Main Memory per node
- Maximum 128 nodes
- Maximum 1024 CPUs, max 9,2 TFLOPS Peak Perf.
- Ultra-High bandwith shared memory subsystem
- Internode crossbar Switch
- 1 TB/s maximum interconnect bandwidth per system

Empowered by Innovation NEC

SX-6+ Technology Breakthrough

- single chip vector processor
- unprecedented memory bandwidth per chip:

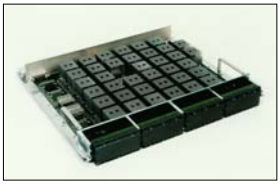
  up to 36 Gbytes/s
- low latency between execution control and vector pipes



Empowered by Innovation NEC

NEC Vector CPU *Evolution*

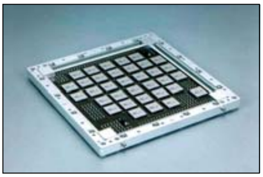
SX - 4



457 x 386 mm

2 GFLOPS at 8.0 ns
0.35μ CMOS
37 Chips

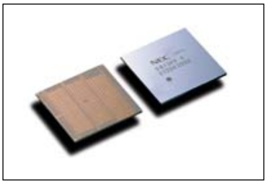
SX- 5



225 x 225 mm

8 GLOPS at 4.0 ns
0.25μ CMOS
32 Chips

SX- 6+



11,0 x 11,5 mm

9 GFLOPS at 1.77 ns
0.15μ CMOS
Single Chip Processor

Empowered by Innovation NEC

NEC Memory *Evolution*

SX - 4

457 x 386 mm

up to 32 Mb SDRAM
max. 8 Modules/Card
256MB / Card
1 Card/CPU

SX- 5

457 x 386 mm

64 - 128 Mb SDRAM
max. 16 Modules/Card
up to 8 GB / Card
1 Card/CPU

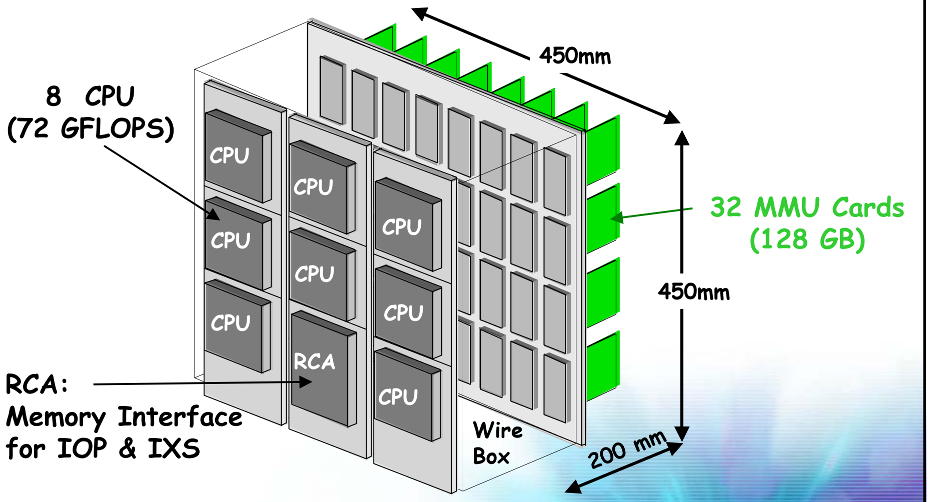
SX- 6+

105 x 176 mm

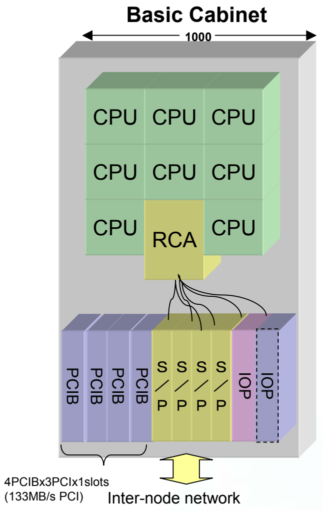
4GB / Card
512 Mb DDR-SDRAM
4 Cards/CPU
= 16 GB/CPU

Empowered by Innovation NEC

CPU-Memory Connection full node SX-6+

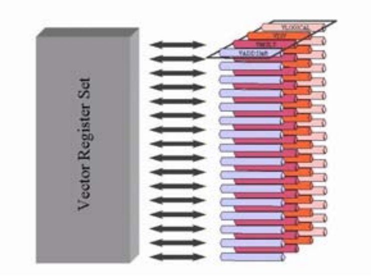


Single Node Small Configuration SX-6



Vector Unit Architecture SX-5

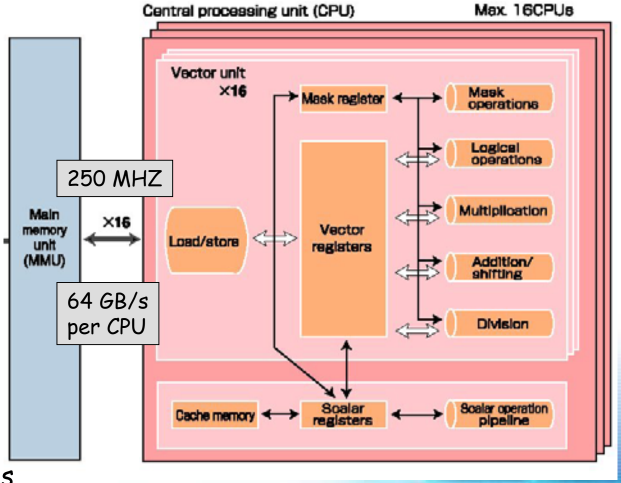
- Multiple Vector Parallel Pipelines
- 64 (32) Pipelines
  - Add-Shift x 16 (8)
  - Multiply x 16 (8)
  - Logical x 16 (8)
  - Divide x 16 (8)



- Each Instruction Uses 16 Pipelines
  - Automatic Hardware Parallelism
- Concurrent Pipeline Set Operation

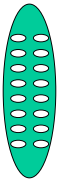
Empowered by Innovation NEC

SX-5 Vector Unit Block Diagram



250 MHZ

64 GB/s per CPU



16-set Vector Pipelines 16 results/clock

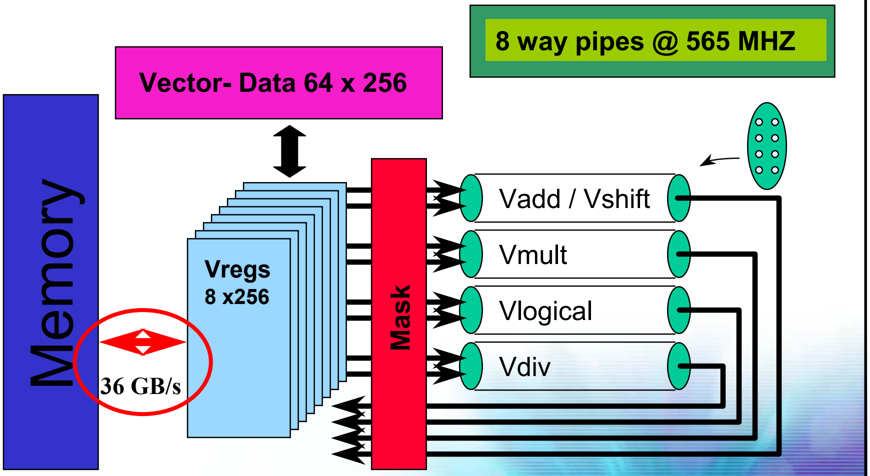
HLRS= 8 !

cycle = 4 ns
1 Func. U. = 8*250 MHZ= 2 GFLOP's
2 Func. U. conc. = 4 GFLOP's

Empowered by Innovation NEC

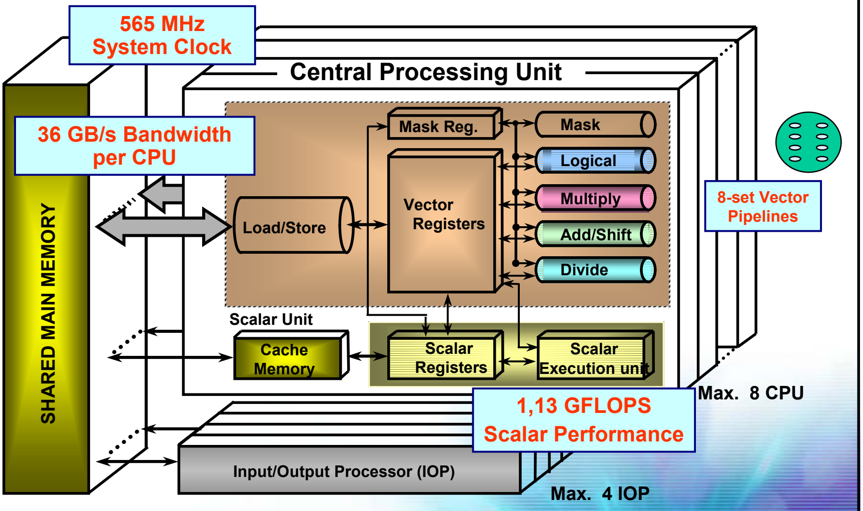
SX-6+ CPU

One Functional Unit : 565 Mhz x 8 = 4,5 GFLOPs



Empowered by Innovation NEC

SX-6+ Processor Architecture



One Functional Unit : 565 Mhz x 8 = 4,5 GFLOPs

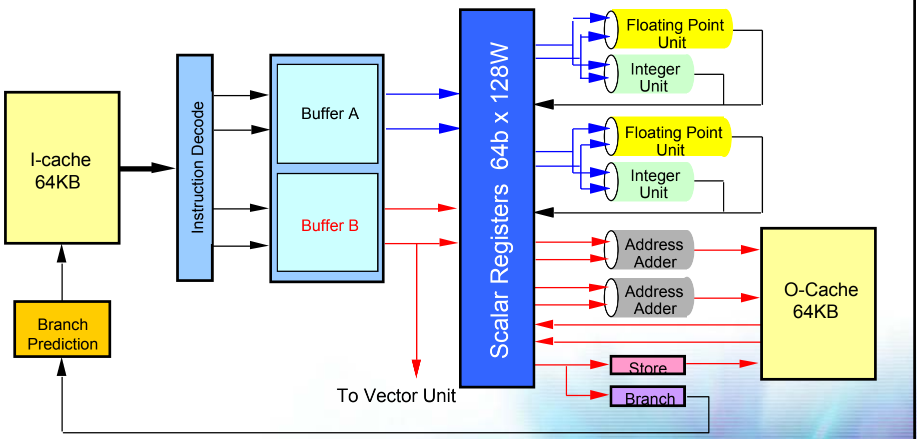
Empowered by Innovation NEC

Scalar CPU: SX-5/6

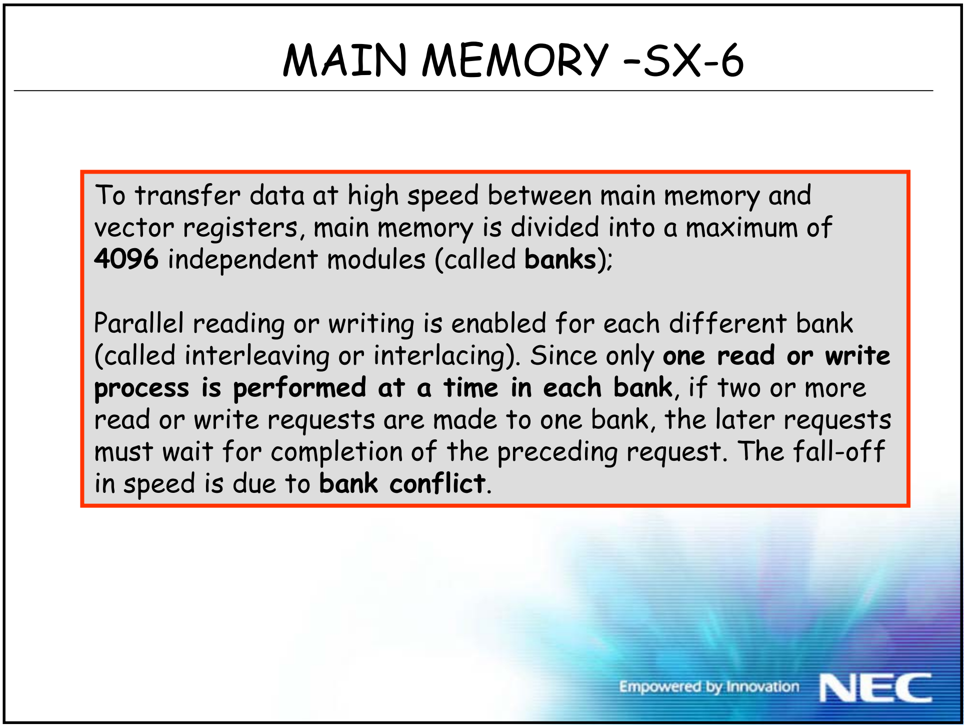
- is more important than one would assume!
- features
  - 2 x 64 kByte Cache (Instructions, Data)
  - 8 kByte Instruction Buffer
  - 128 64-bit registers
  - 4 instructions per cycle
  - instruction reordering (data flow control)
  - branch prediction
  - 2 fmult/fadd/fdiv pipes, 2 integer pipes
  - double word load (SX-4: single)
  - 565 MHZ -> 1,13 GFLOPs Peak

Empowered by Innovation NEC

SX-5/6 Scalar Unit Block Diagram



Empowered by Innovation NEC

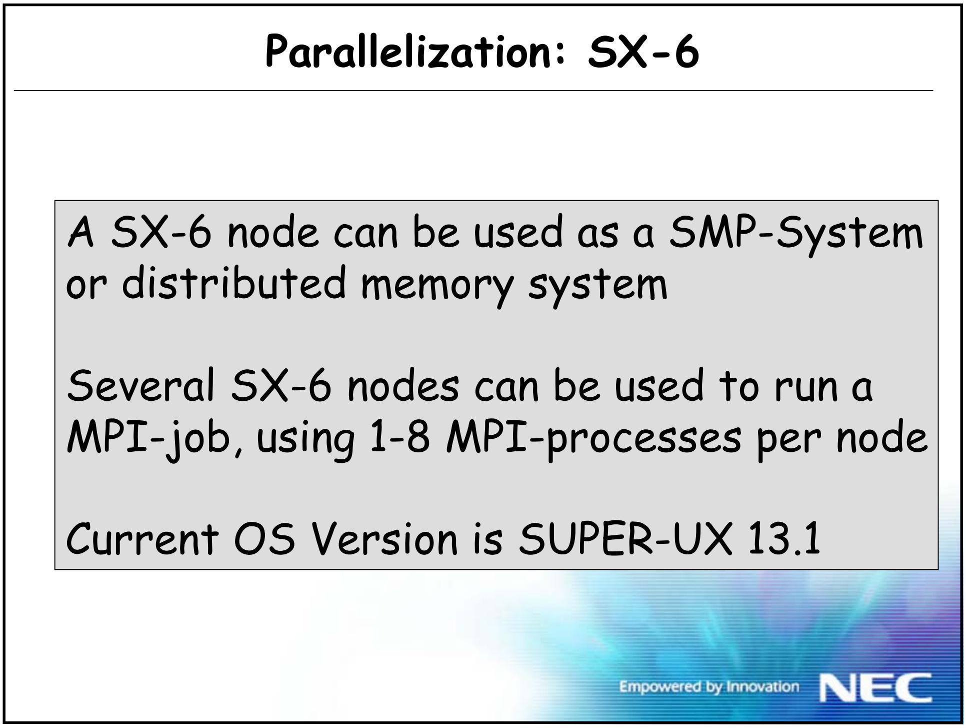


MAIN MEMORY –SX-6

To transfer data at high speed between main memory and vector registers, main memory is divided into a maximum of **4096** independent modules (called **banks**);

Parallel reading or writing is enabled for each different bank (called interleaving or interlacing). Since only **one read or write process is performed at a time in each bank**, if two or more read or write requests are made to one bank, the later requests must wait for completion of the preceding request. The fall-off in speed is due to **bank conflict**.

Empowered by Innovation NEC



Parallelization: SX-6

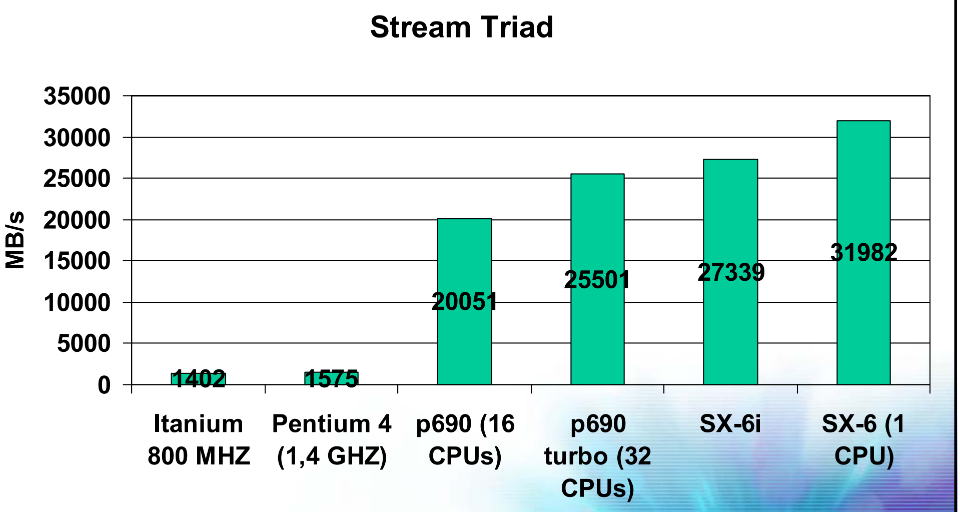
A SX-6 node can be used as a SMP-System or distributed memory system

Several SX-6 nodes can be used to run a MPI-job, using 1-8 MPI-processes per node

Current OS Version is SUPER-UX 13.1

Empowered by Innovation NEC

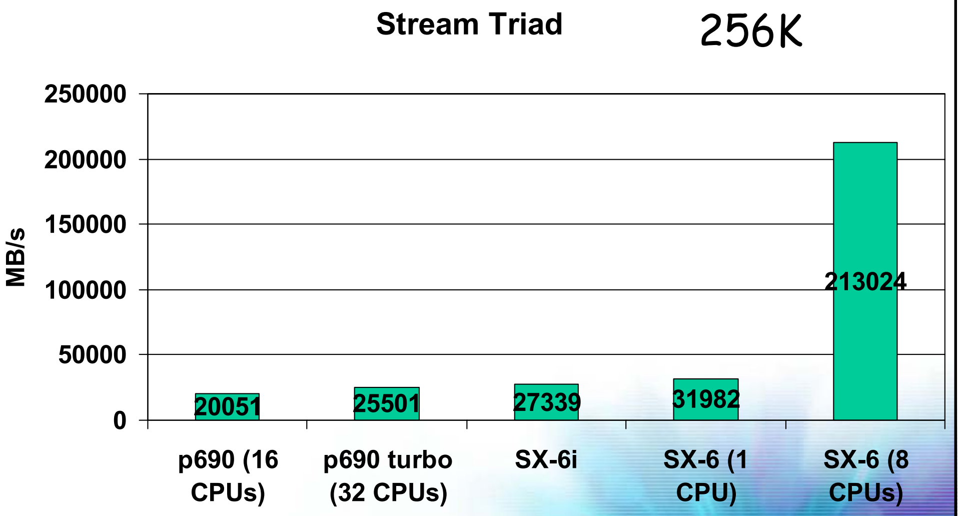
Memory Bandwidth I



Peak= 32 GB/s

Empowered by Innovation NEC

Memory Bandwidth II



256K

Empowered by Innovation NEC

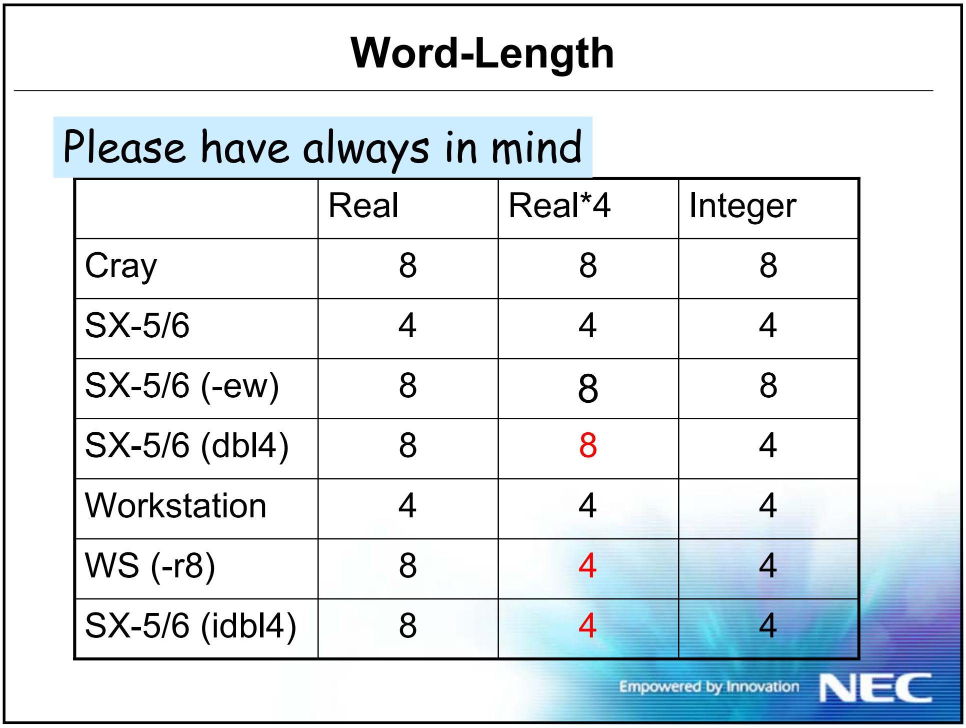
Basics on Vectorization

Empowered by Innovation NEC

IEEE Format (float0) !!!

- 4 Byte:
  - about 7 digits
  - 10^{-38} - 10^{38}
- 8 Byte:
  - about 16 digits
  - 10^{-308} - 10^{308}
- 16 Byte:
  - about 32 dig
  - 10^{-308} - 10^{308}
  - **not vectorizable!**

Empowered by Innovation NEC

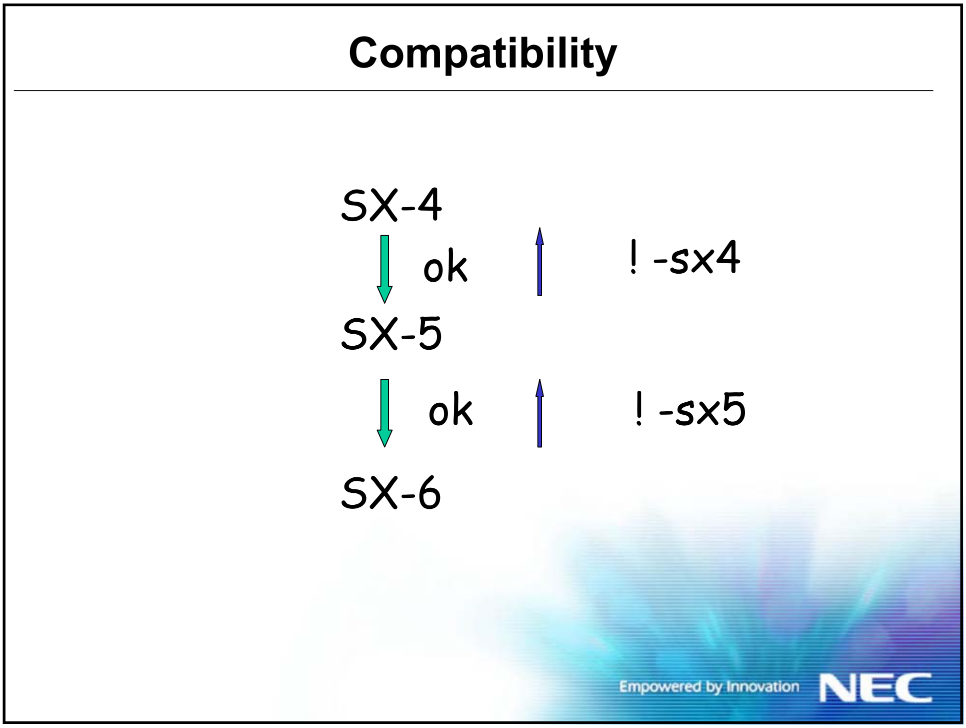


Word-Length

Please have always in mind

	Real	Real*4	Integer
Cray	8	8	8
SX-5/6	4	4	4
SX-5/6 (-ew)	8	8	8
SX-5/6 (dbl4)	8	8	4
Workstation	4	4	4
WS (-r8)	8	4	4
SX-5/6 (idbl4)	8	4	4

Empowered by Innovation NEC



Compatibility

SX-4

ok ! -sx4

SX-5

ok ! -sx5

SX-6

Empowered by Innovation NEC

Basics on vectorization

First very important principle:
segmentation of operations and pipelining

Empowered by Innovation NEC

Segmentation & Pipelining

- Operations are decomposed into segments
- Example: add Floating point number

1.14e9 - 2.78e8

1.14e9 - 0.278e9

0.862e9

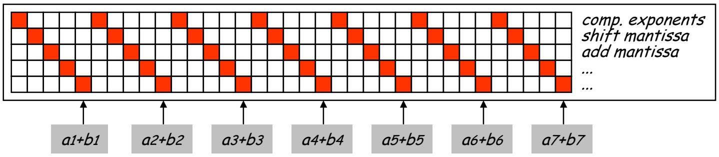
8.62e8

▶ compare exponent
▶ Shift mantissa
▶ Add mantissa
▶ Select exponent and normalize

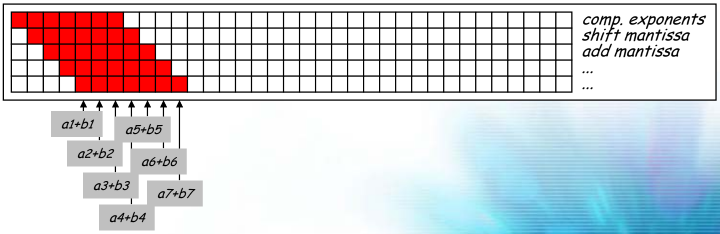
Empowered by Innovation NEC

Pipelining (cont.)

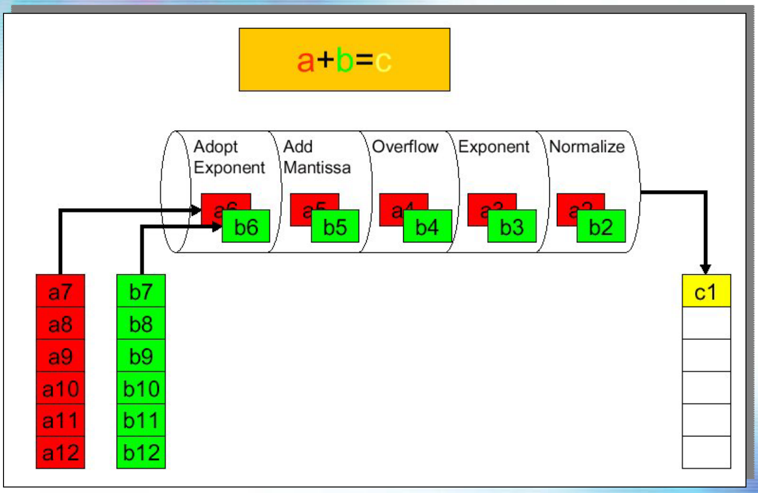
- no pipelining:



- with pipelining:

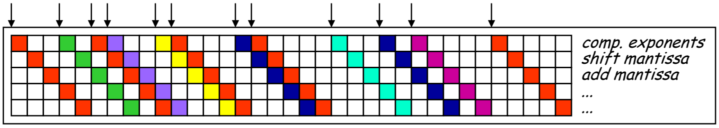


Segmentation Pipelining

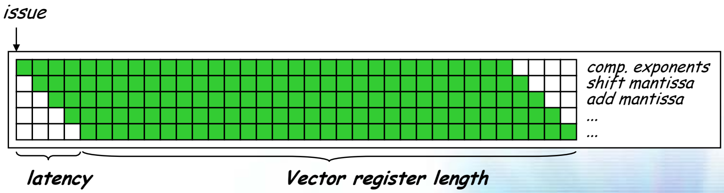


Pipelining (cont.)

- Superscalar pipeline (RISC) :



- Vector pipeline:



Simple Example: Data Parallelism

- 'Vector Loop': Data Parallel
- Independent data

F77:

```
Real a(n), b(n), c(n)

do i = 1, n
  a(i) = b(i) + c(i)
end do
```

F90:

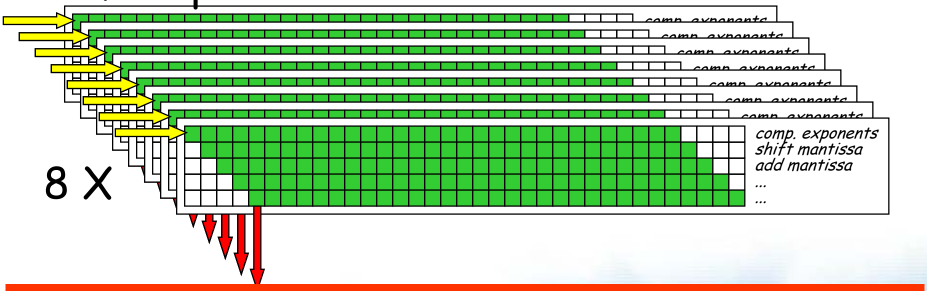
```
Real, dimension(n): a, b, c

a = b + c
```

N=256, Loop will be divided in 8 parts

Pipelining (cont.)

This is the basics of vectorization !!

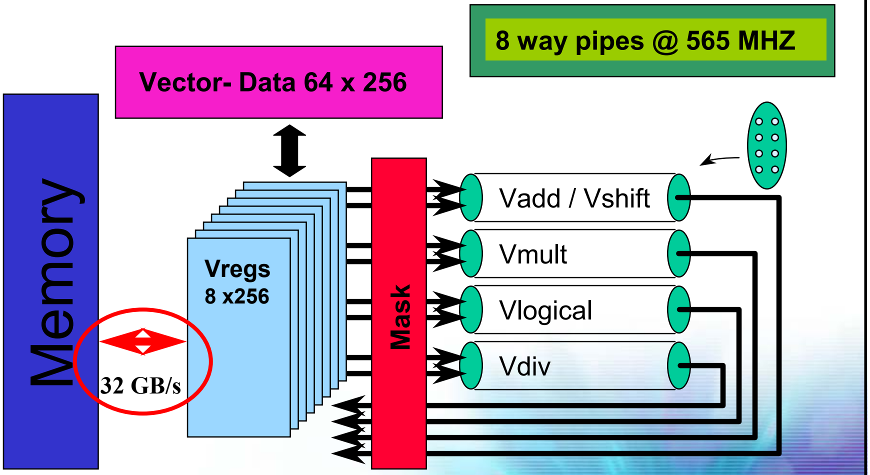
8-fold parallel on SX-6



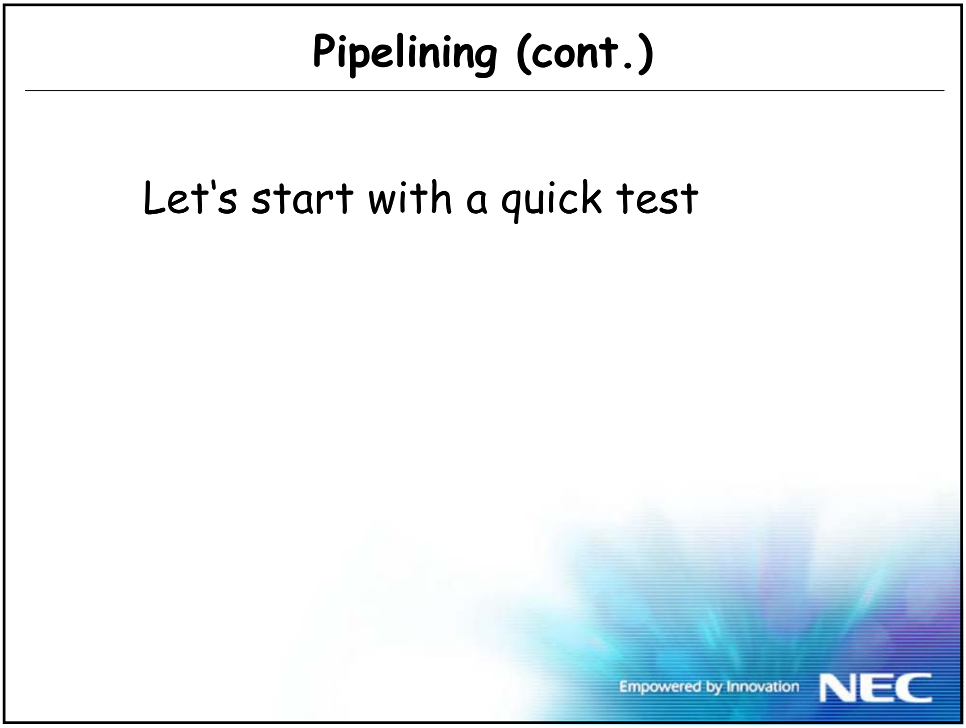
Optimizations means: Keep these pipelines busy

Empowered by Innovation NEC

SX-6 CPU

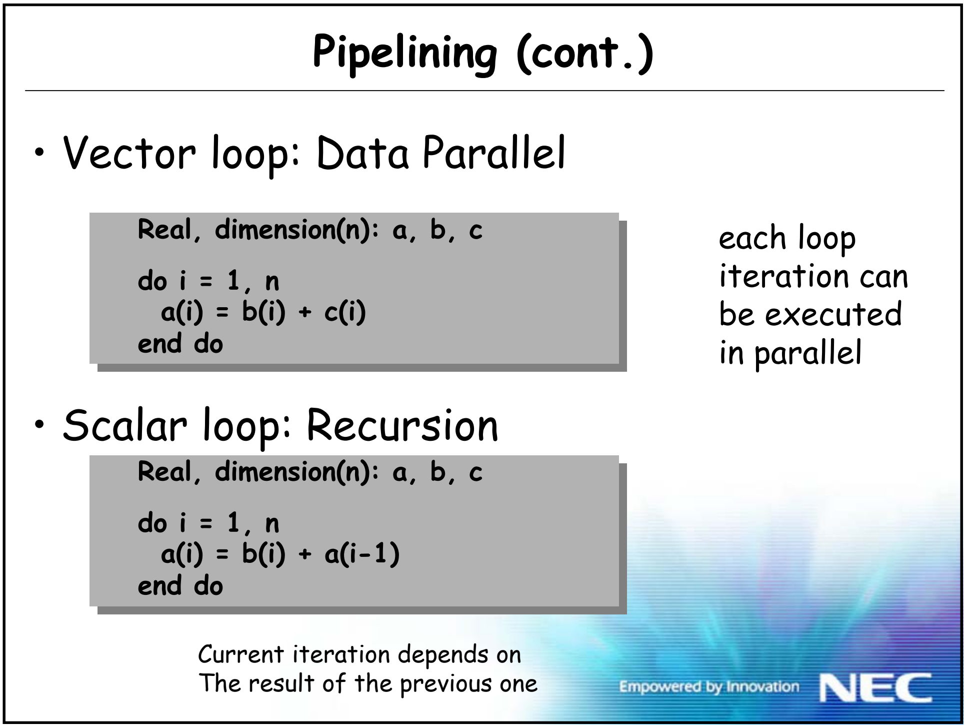


Empowered by Innovation NEC



Pipelining (cont.)

Let‘s start with a quick test



Pipelining (cont.)

- Vector loop: Data Parallel

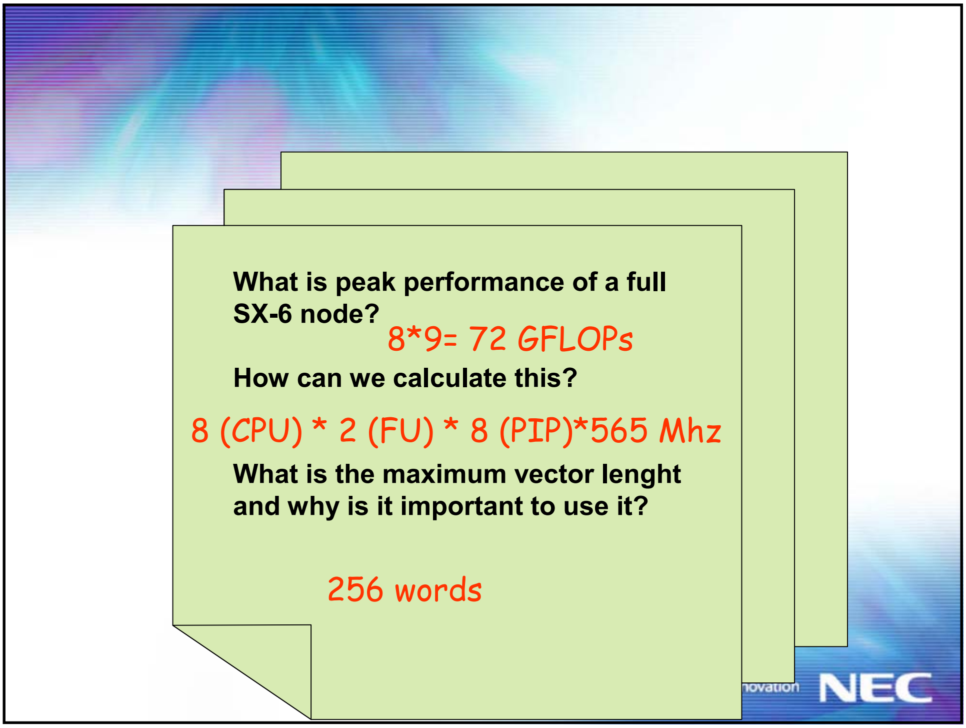
```
Real, dimension(n): a, b, c

do i = 1, n
  a(i) = b(i) + c(i)
end do
```

each loop iteration can be executed in parallel

- Scalar loop: Recursion

```
Real, dimension(n): a, b, c

do i = 1, n
  a(i) = b(i) + a(i-1)
end do
```

Current iteration depends on
The result of the previous one



What is peak performance of a full SX-6 node?

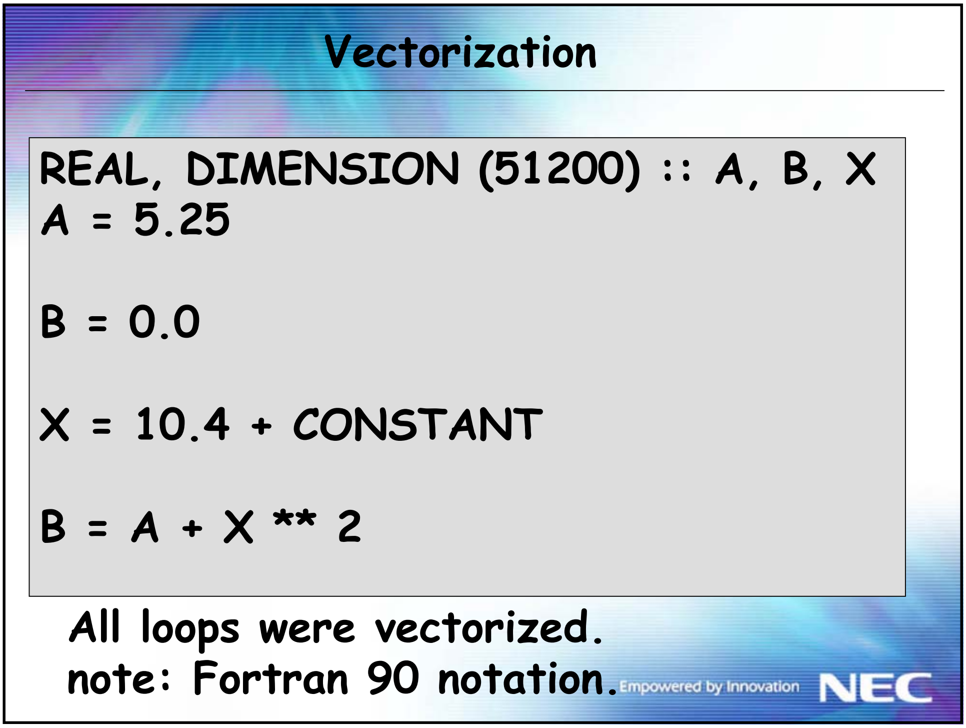
8*9= 72 GFLOPs

How can we calculate this?

8 (CPU) * 2 (FU) * 8 (PIP)*565 Mhz

What is the maximum vector lenght and why is it important to use it?

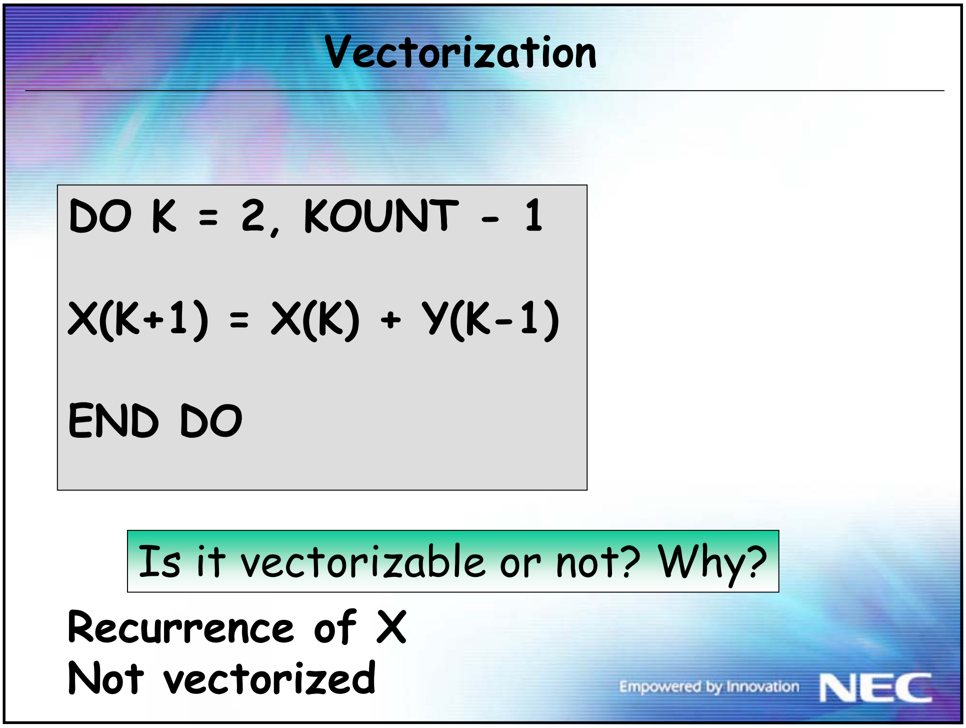
256 words



Vectorization

```
REAL, DIMENSION (51200) :: A, B, X
A = 5.25

B = 0.0

X = 10.4 + CONSTANT

B = A + X ** 2
```

All loops were vectorized.
note: Fortran 90 notation.



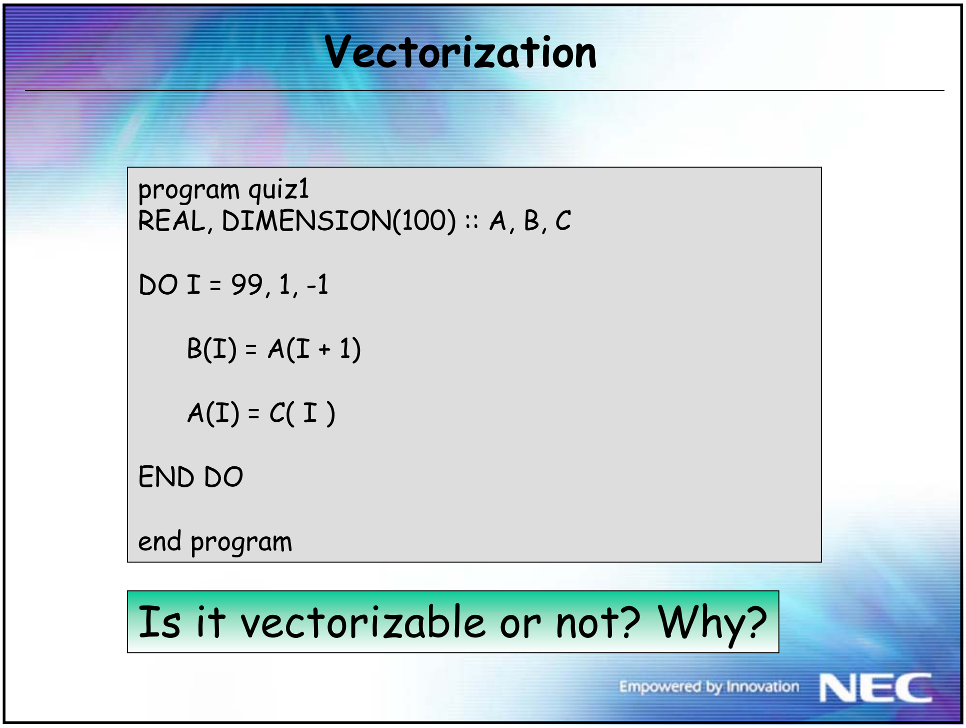
Vectorization

```
DO K = 2, KOUNT - 1

X(K+1) = X(K) + Y(K-1)

END DO
```

Is it vectorizable or not? Why?

Recurrence of X
Not vectorized

Empowered by Innovation NEC

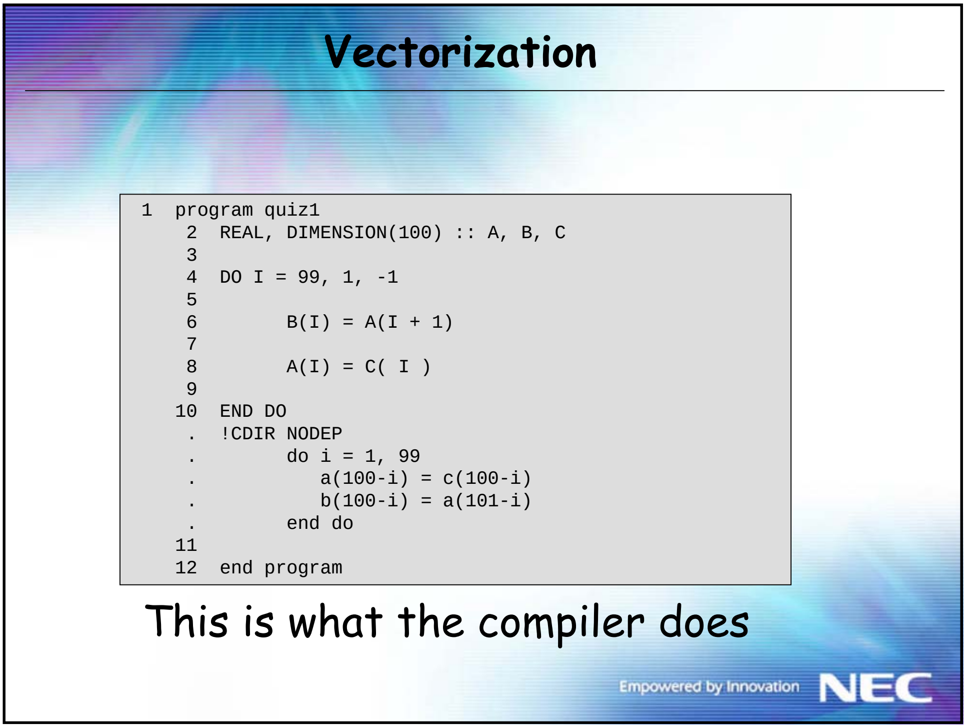


Vectorization

```
program quiz1
REAL, DIMENSION(100) :: A, B, C

DO I = 99, 1, -1

     B(I) = A(I + 1)

     A(I) = C( I )

END DO

end program
```

Is it vectorizable or not? Why?

Empowered by Innovation NEC

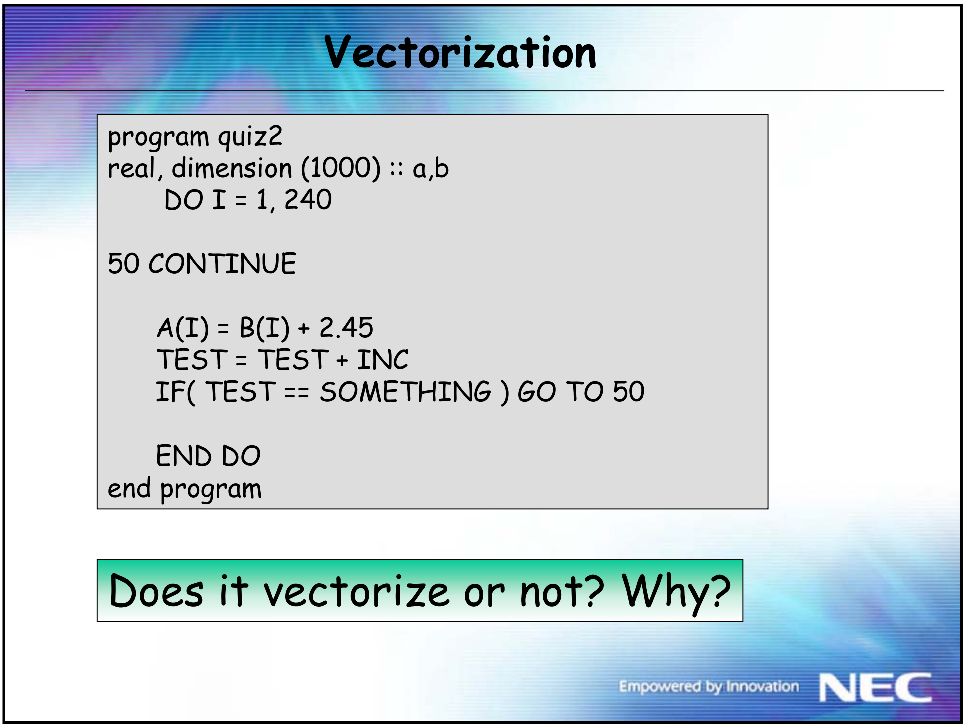


Vectorization

```
program quiz1
REAL, DIMENSION(100) :: A, B, C

DO I = 99, 1, -1

      B(I) = A(I + 1)

      A(I) = C( I )

END DO
!CDIR NODEP
      do i = 1, 99
         a(100-i) = c(100-i)
         b(100-i) = a(101-i)
      end do

end program
```

This is what the compiler does

Empowered by Innovation NEC

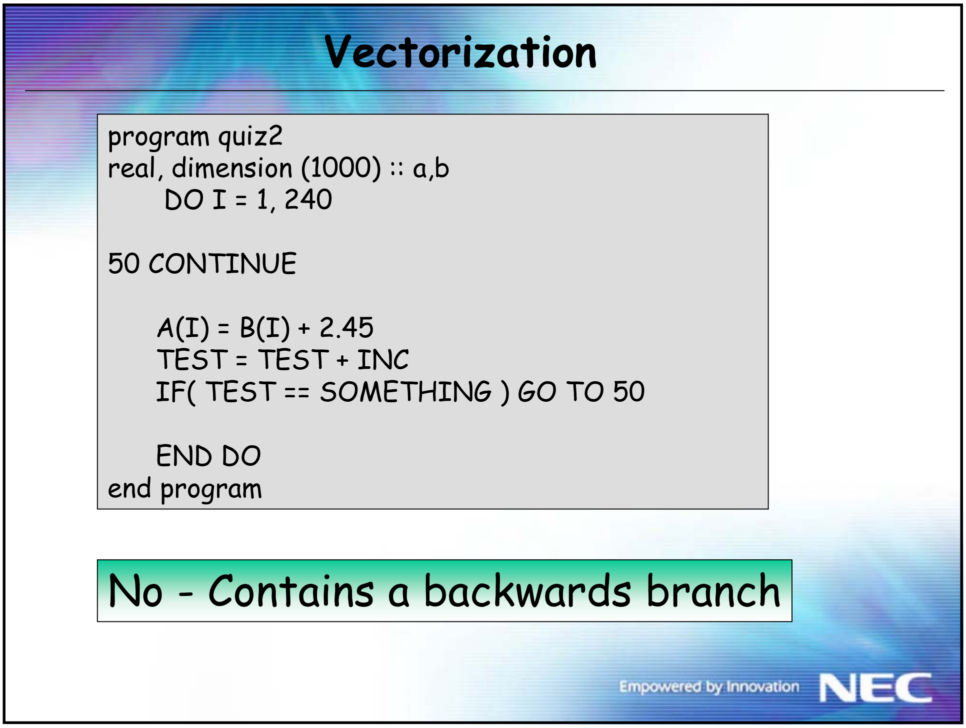


Vectorization

```
program quiz2
real, dimension (1000) :: a,b
     DO I = 1, 240

50 CONTINUE

    A(I) = B(I) + 2.45
    TEST = TEST + INC
    IF( TEST == SOMETHING ) GO TO 50

    END DO
end program
```

Does it vectorize or not? Why?

Empowered by Innovation NEC

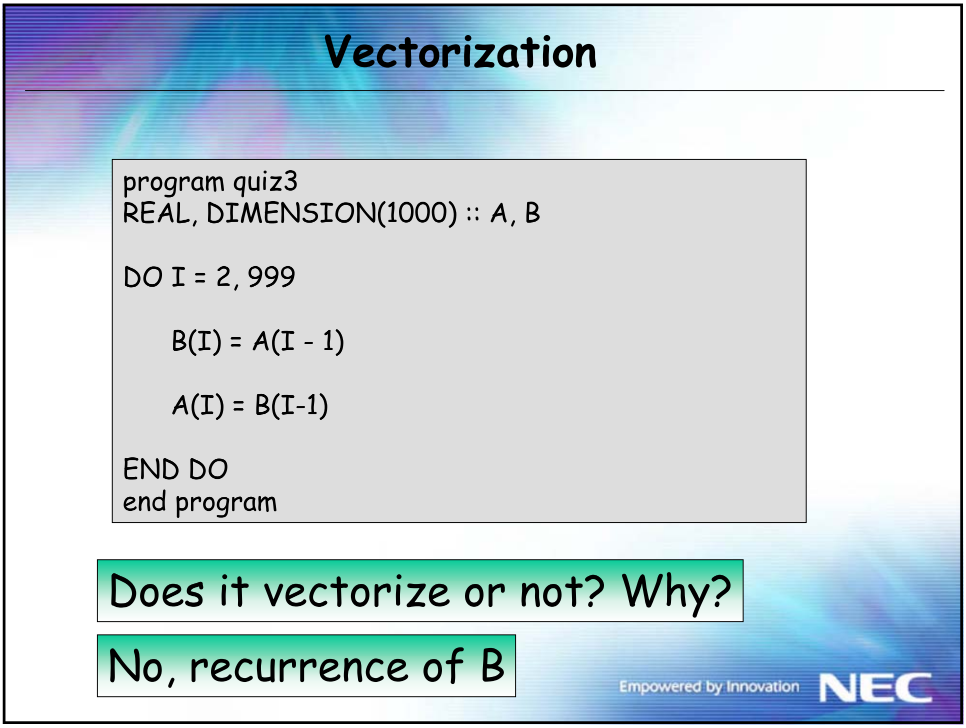


Vectorization

```
program quiz2
real, dimension (1000) :: a,b
      DO I = 1, 240

50 CONTINUE

     A(I) = B(I) + 2.45
     TEST = TEST + INC
     IF( TEST == SOMETHING ) GO TO 50

     END DO
end program
```

No - Contains a backwards branch

Empowered by Innovation NEC



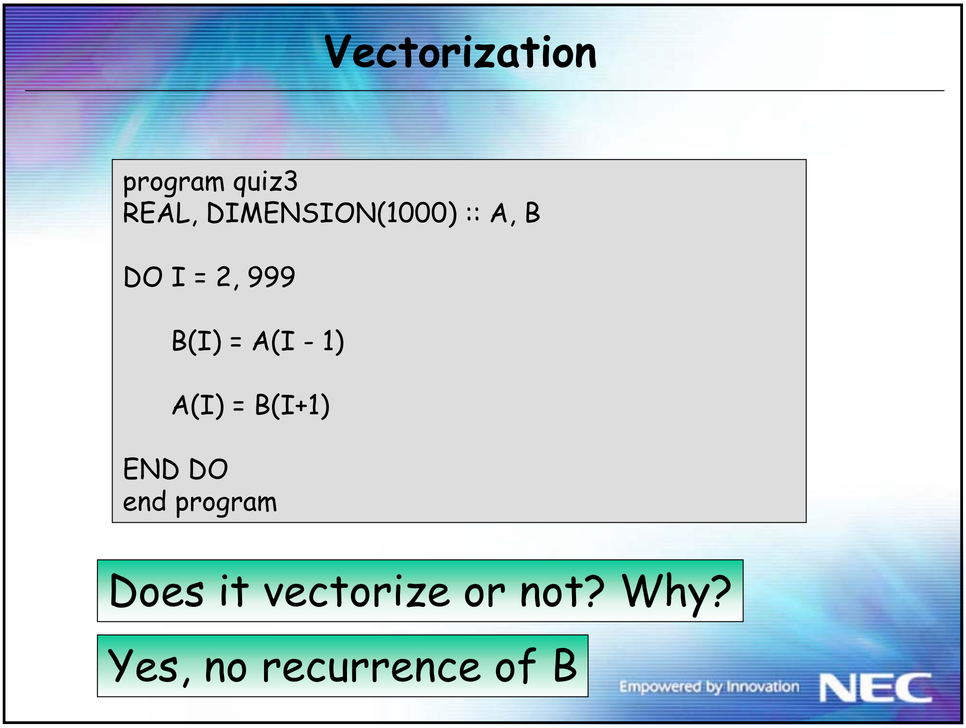
Vectorization

```
program quiz3
REAL, DIMENSION(1000) :: A, B

DO I = 2, 999

     B(I) = A(I - 1)

     A(I) = B(I-1)

END DO
end program
```

Does it vectorize or not? Why?

No, recurrence of B

Empowered by Innovation NEC



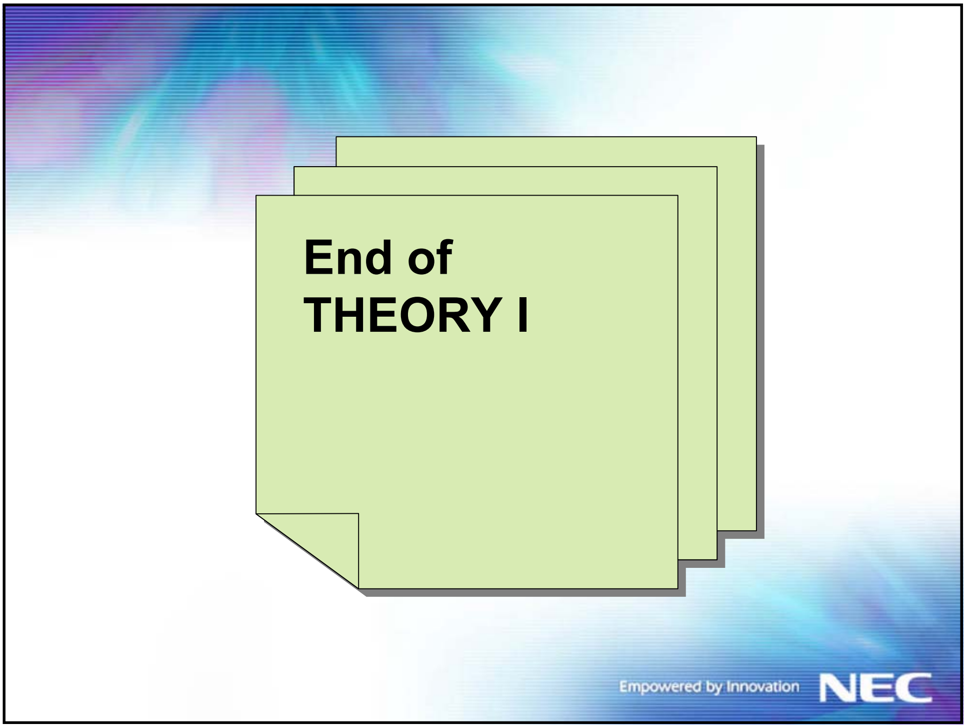
Vectorization

```
program quiz3
REAL, DIMENSION(1000) :: A, B

DO I = 2, 999

     B(I) = A(I - 1)

     A(I) = B(I+1)

END DO
end program
```

Does it vectorize or not? Why?

Yes, no recurrence of B

Empowered by Innovation
NEC



End of
THEORY I

Empowered by Innovation
NEC