

## **Past, present and future of microprocessors**

*F. Anceau*

*Conservatoire National des Arts et Métiers*

*292, rue Saint Martin, Paris cedex 03, France*

*tel: +33 (0)1 40 27 24 33, fax: +33 (0)1 40 27 28 45,*

*e-mail: anceau@cnam.fr*

### **Abstract**

Microprocessors are one of the most important technical phenomena of the end of this century. Since twenty-five years their computing power and complexity have increased at sustained rates. Microprocessors are increasingly playing a major role in modern society. The "invisible" ones, used for controlling and monitoring machine tools, cars, aircraft, consumer electronics and other equipment are the most numerous. They are gradually changing the relationship we have with these devices. It is interesting to show that this is a "market-pull" rather than a "technology-push" phenomenon. The design of new chips thus represents a continuous challenge for the engineers and technologists striving to give the market the products it requires, and which are generally planned a long time before they actually appear. Monolithic microprocessors are overtaking all kinds of computers. Minicomputer lines were absorbed during the 80's, main-frame lines during the 90's and probably super-computers by the beginning of the next century. During this extraordinary evolution, these devices have used all the technical innovations which had been conceived for the previous generations of computers. The future of these devices is very challenging. To keep the evolutionary rate of computing power and binary code compatibility, completely new execution techniques will need to be invented, eventually leading to the break-down of the physical quantum barrier around 2010.

### **Keywords**

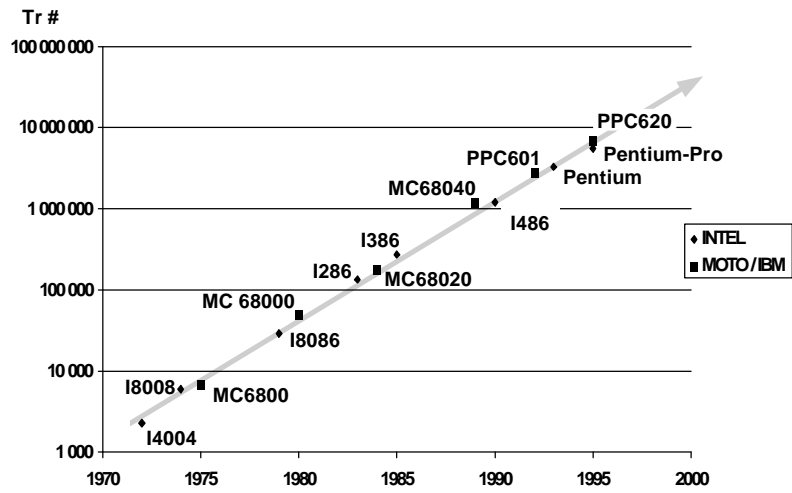
Microprocessors architecture, microprocessors design, microprocessors evolution.

## **1 INTRODUCTION**

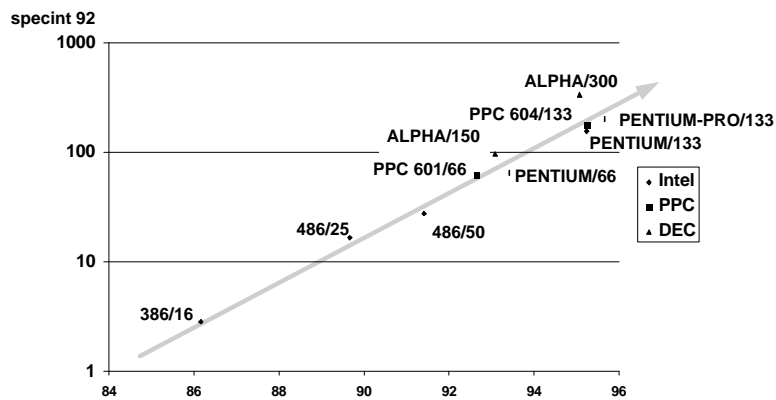
The story of microprocessors (that we could call VLSI processors) began in 1972 with the Intel 4004 which was designed to build calculators, but which was also used in many various applications. Since this date microprocessors have become powerful computers which are taking over all the other ranges of computers. Microprocessors are one of the most important technical phenomena of the end of this century.

## **2 A SUSTAINED EVOLUTION RATE**

Since more than a quarter of century the evolution of microprocessors is continuing at a sustained rate. The complexity of these devices increase with a mean rate of 37% per year and their performances (measured with a common evaluation tool) increase with the same rate.

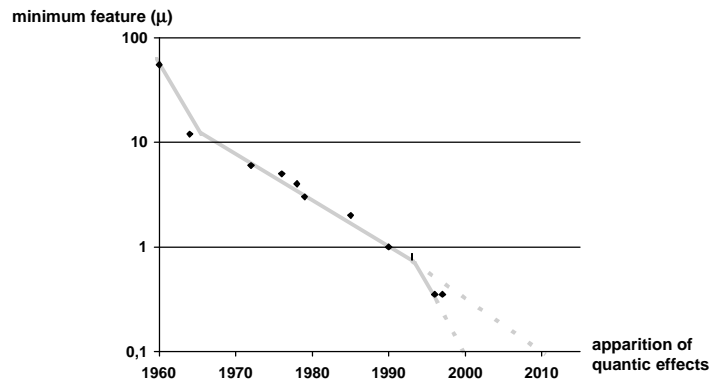


**Figure 1** Evolution of the complexity of microprocessors.



**Figure 2** Evolution of the performance of microprocessors.

The evolution of the complexity of these devices is very linear showing the advance of the technological evolution. The minimum feature size decreases regularly and the size of the chip increases.

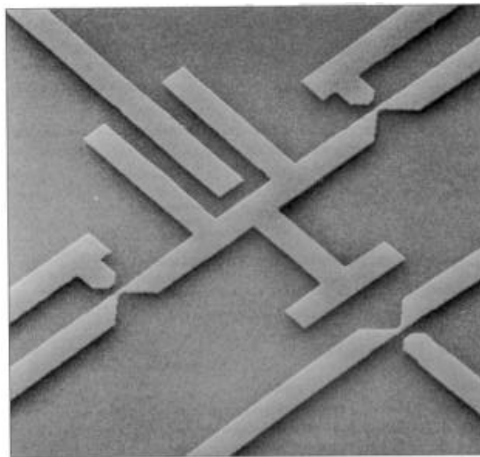


**Figure 3** Technological evolution.

Everybody (including the experts) are predicting a decreasing of this evolution rate because the extraordinary difficulty to manufacture such devices. But this evolution is surprising because since three years it appears that the evolution rate is increasing. The current feature size of the better technology is around  $0.35\mu$  instead  $0.45\mu$  given by continuing the previous evolution rate.

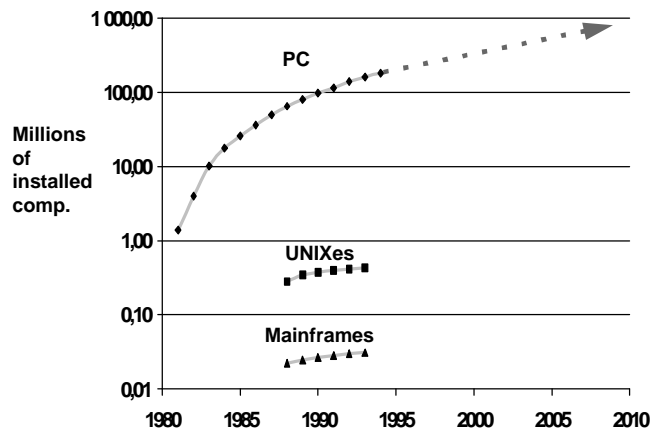
At this speed of evolution, the quantic phenomena, which are occurring around  $0.1\mu$ , will occur before 2010 making the "classic" MOS transistors no longer usable.

The economical pressure is so high that we will have a good chance to break down the quantic barrier before this date. Already some quantic transistors are beginning to run in several research laboratories.



**Figure 4** Single electron memory from Hitachi Cambridge Lab.

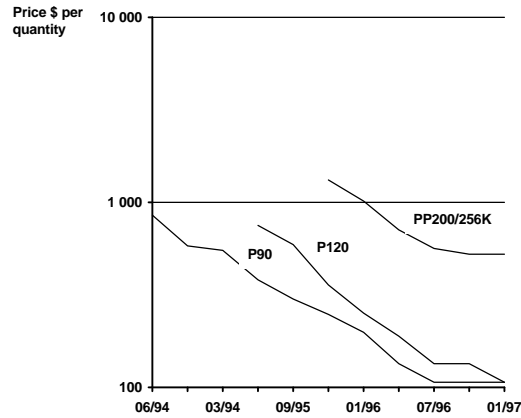
From an economical point of view, microprocessors have contributed to the democratization of computers. They have contributed to decrease the price of computers by dropping the cost of their electronic. They were at the origin of the personal computer phenomena. The market of these kind of computers is now dominant in the computer industry. The size of the series of these machines is far above the best series of mainframe or UNIXes computers. The personal computer industry is now a mass industry of commonality products.



**Figure 5** Installed base of families of computers.

Around 1985 the VLSI processor evolution had moved from a market pushed by the technological evolution to a market pulled by the requirements of the end-users which always want to use the last software packages. To do that, they have to buy more powerful (personal) computers to run them. The size of this market is so high that a large amount of money is given for technological improvements and its large series make possible mass production.

As any other mass-production VLSI devices, the price of microprocessors follows a classic curve starting at an introductory price around \$1000 decreasing in few years down to \$100. It is always difficult to imagine that the price of such new extraordinary devices will fall down soon to the price of commonality goods.



**Figure 6** Price evolution of microprocessors.

#### 4 LOOKING TO THE FUTURE

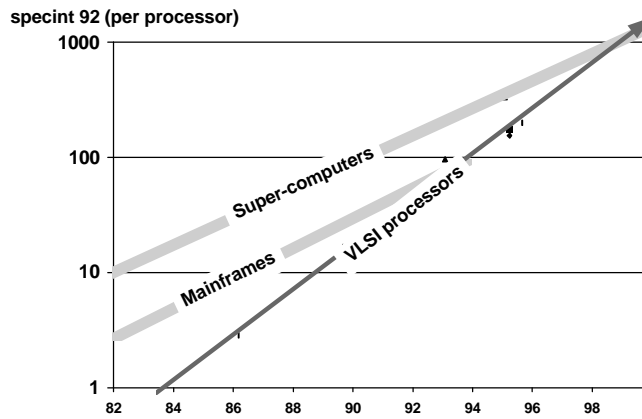
The prediction capability of these curves is very good. It is possible to predict the computing power and the size of a future VLSI processor with a good accuracy. However the prediction of its architecture principle is more difficult to do because the segment representing the future in this curve represent the challenge put to the designers of microprocessors.

We can extend up to 2010 (before the quantic barrier occurs) the curves of complexity and performance of microprocessors. The result is very impressive. At this time we will have processors with a complexity around 1Gtr which will run at a speed around 100G instr./s (100 Gips).

It looks difficult to trust such predictions. But if we make a step back to 1986, at this time nobody was trusting that a 0.5 Gips microprocessors would be available ten years later. To-days in some laboratories they are already microprocessors running faster than 1Gips.

#### 5 VLSI PROCESSORS ARE ABSORBING ALL THE RANGE OF COMPUTERS

The computers which are build using microprocessors are absorbing all the range of computers. They have already absorbed the range of minicomputer during the 80's and they are absorbing the range of mainframes during the 90's. Around 1993, the microprocessors have taken over the mainframes range in term of complexity and performance. Probably they will absorb the range of supercomputers at the beginning of the next century.



**Figure 7** Microprocessors are absorbing the range of computers.

The necessary large series of microprocessors (more than a million per year to be economically viable) are a lead to reduce the number of different models. As example, the series of the VLSI version of the mainframe or UNIXes computers are not so large to economically compete with the microprocessors of the personal computers.

## 6 MICROPROCESSORS ARE CHANGING OUR ENVIRONMENT

Microprocessors are increasingly playing a major role in the modern society. They can be put in two classes: The "visible" ones used to build the different classes of computers, mostly the personal ones, and the "invisible" ones used for controlling and monitoring machine tools, cars, aircraft, consumer electronics and many other electronic equipment. The class of invisible microprocessors is the most important one in term of the number of microprocessors in use. They are gradually changing the relationship we have with these devices.

We can ask us where this phenomenon is leading us. The microprocessors are changing insidiously our environment by making intelligent many objects which were previously simple. Putting a drop of intelligence into many devices is now possible for a little amount of money. We are only at the beginning of this deep cultural evolution.

## 7 THE COMPUTER HERITAGE

During their extraordinary evolution, microprocessors are using all the technical innovations which has been conceived for the previous generations of computer since the beginning of their story, like: pipeline architecture, cache, super-scalar architecture, RISC, out-of-order execution, branch prediction, etc.... All these features are used today in modern microprocessors which are probably the most advanced processors available on the market. They become the followers of the computer story. To keep the rate of their evolution it will be necessary to discover new kind of acceleration techniques.

Microprocessors become central components of the computer design. Building a processor with MSI technology is now completely obsolete and far outside the economical optimum.

Now, all the new computers are based on the use of microprocessors (e.g. CRAY T3E super-computer which is using several ALPHA microprocessors).

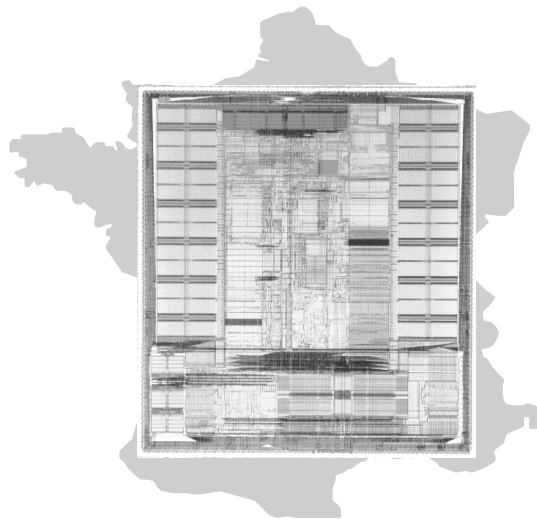
## 8 MICROPROCESSORS AS VLSI CIRCUITS

Microprocessors are obviously VLSI circuits. In such devices the chip define an internal world which is far smaller and faster than the external one. The capacitance are measured in ff, current in  $\mu\text{a}$ , size in  $\mu\text{m}$ , delay in ps. The internal gates of such an integrated circuit are running far faster than the MSI ones.

The cost of passing from the internal world of a VLSI chip to the external world is very high. Multi-level amplifiers and a geometrical translator, provided by the package, are necessary to make this translation. The difference between the internal world and the external one is comparable to driving electromechanical devices from an electronic board.

This cost leads VLSI designers to put as much as possible functional blocks inside a single chip instead of using a multi-chip architecture which multiplies the (costly) interfaces. This feature leads to increase the integration level and the complexity of the chip.

Another important parameter is the distance inside the chip. At its scale a chip is a very large world! We can compare it to a country 600km wide (larger than Switzerland and Belgium together) with roads 10m large. Such a large area must be organized by architectural (urbanistic!) principles. As example, the different blocks constituting the chip layout must fit together as much as possible. Their shape must be studied to obtain the best imbrication as possible.



**Figure 8** Chip as a country.

The cost of transferring information from one end of the chip to another is very high. The relationship between the different blocks must be carefully studied to increase local exchange and decrease long distance communication.

## 9 PROCESSORS AS INTERPRETORS

A processor can be seen as an interpreter of its instruction language. The interpretation algorithm is executed by hardware and eventually also by a microprogram. Many tricks are used to speed up the execution of this algorithm, mostly by increasing overlapping in its execution.

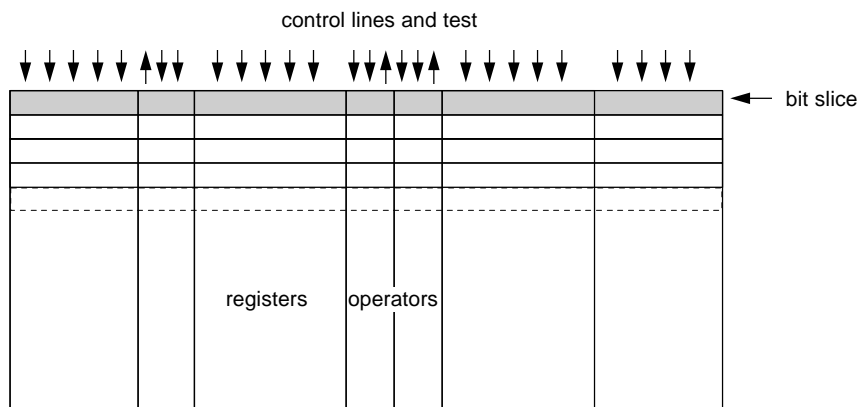
All processors are executing interpretation algorithm to perform instruction execution. Some kind of VLSI circuits are executing other kind of algorithm like signal processing (e.g. digital filtering), data communication, mathematical co-processing.... They are not called microprocessors but their internal structure is very similar.

The functional organization of a processor is basically split into two main blocks called: control section and data path. The specification of these two blocks come from the decomposition of the interpretation algorithm into its control structure and the set of its operations. The control structure of this algorithm is used to specify the control section and its set of operations is used to specify the data path.

Since the Motorola 6800 microprocessor, launched in 1974, these two functional blocks are implemented by two separated pieces of layout designed with specific techniques.

## 10 DATA-PATH DESIGN

The data path is made of operators, buses, registers. Very powerful techniques are used to design its layout. The basic techniques to design data path is the use of the bit slice layout strategy. A rectangular layout is obtained by the juxtaposition of layout strips representing a single bit of data for all the data path. These layout strips are constituted by the assembly of cells which have a fixed width. The buses are included in the layout. The connection of the cells to make such a strip is obtained by their simple abutment.



**Figure 9** Layout organization of a data-path.

The layout of data path is obtained by using CAD tools called silicon assemblers. These tools perform the assembly of the different cells constituting the data path. Some of these tools perform automatic resizing of the transistors and add a network for power supply and a line of amplifiers for the control lines which are located on the side of the data path layout.

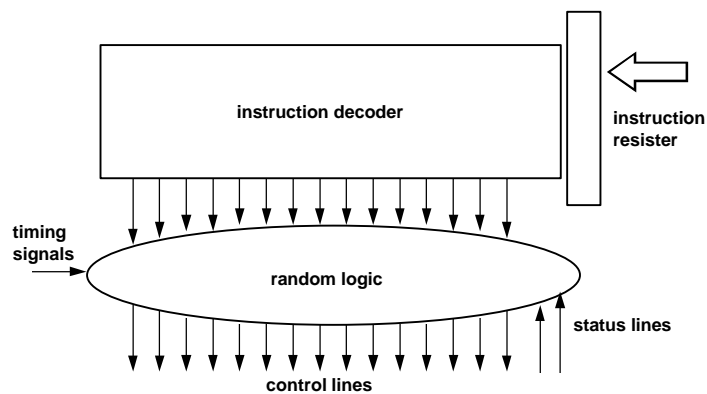
The resulting layout is very dense and very optimized. Data-path is probably the best layout piece of a microprocessor. Its area is between a quarter and half of the full area of the microprocessor.

## 11 CONTROL-SECTION DESIGN

Unfortunately, there are no global techniques to design the control section. It is not really a single block of layout but the assembly of several blocks like ROM, PLA, decoders and random logic. The layout of control section is designed by the traditional techniques of placement and routing.

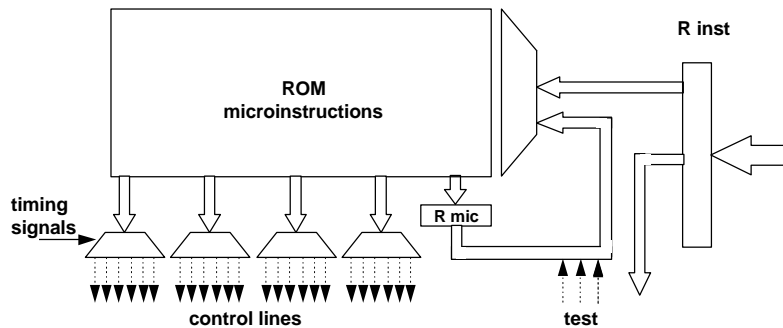
There are several architectural approaches to design control section. Some of them are inspired from the traditional computers. But the opportunity to design specific blocks at VLSI level opens up new interesting techniques.

The control section of a microprocessor can be hardwired or microprogrammed. A hardwired control section is designed around a PLA (Programmable Logic Array) performing the decoding of the instructions. A series of gates are used to validate the result of this decoding by the timing signals provided by the clock circuitry to obtain the commands for the data-path.



**Figure 10** Hardwired control-section.

A microprogrammed control section is designed around a ROM where microinstructions are stored. A microprogram address counter is used to address the ROM. Microinstructions are read from the ROM into a microinstruction register where their different fields are decoded. The resulting commands for the data-path are obtained by validating the outputs of this decoding by the timing signal provided by the clock circuitry.



**Figure 11** Microprogrammed control-section.

The microprogrammed control sections are more adapted to the execution of complex instructions which are using a variable number of clock cycles. They are mostly used by CISC microprocessors.

The hardwired control sections are more adapted to the execution of simple instructions which are using a fixed timing pattern. They are used in RISC microprocessors.

Both these kind of control sections are using blocks of random logic which are designed as strips of cells interconnected by channel routings like in pre-characterized design technique.

## 12 RISC COMPUTERS

In 1975 John Cocke invents the notion of Reduced Instruction Set Computer (RISC) in an IBM laboratory. In a classical computer (called CISC for Complex Instruction set Computer) the use of the instructions is very unbalanced. As examples, the rate of the use of the branch instructions is about 25% whereas the use of the instructions to manipulate character string is far under 1%. By selecting a class of the instructions more frequently used, RISC computers are well balanced and the use of each part of their hardware can be optimized. Several features are specific to RISC processors:

- A large number of registers are used to decrease the memory accesses.
- Few instruction formats are used to simplify instruction decoding
- All the instructions (except for the memory accesses) are executed in the same number of clock cycles simplifying the sequencing of the execution.
- Memory accesses are done by few specific instructions (which perform move operations between memory and registers)

The hardware / software interface of the RISC processors is lower than in the CISC processors. This low interface come from the reducing of the functions performed by the hardware and from the increasing of the complexity of those executed by the software. The programs written for a RISC processor are larger than those for a CISC one, but the simplicity of the RISC instructions allows a very good optimization of the programs by compilers. This optimization and the short clock cycle of the RISC computers give them an higher global performance than the CISC ones.

The hardware of RISC microprocessors is far simpler than those of a CISC one. The control section of a RISC microprocessor is fully hardwired. Its area is about the same as the area of its data-path.

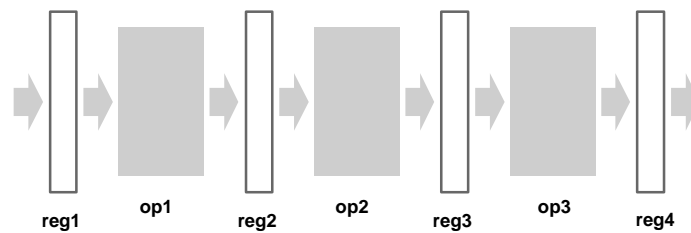
The most popular series of microprocessor is a CISC one (x86) but all the other are RISC ones.

### 13 SPEEDING UP THE EXECUTION

The pressure to increase the performance of the microprocessors led their designers to find new ways to design these machines. All the techniques which are used are based on superposition in the execution of the interpretation algorithm in order to start the execution of a new instruction before terminating the previous ones.

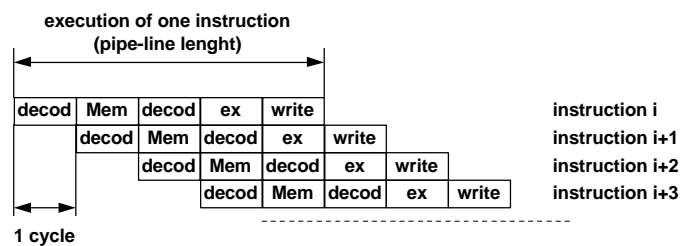
### 14 PIPELINE EXECUTION

The idea of this execution technique is to split the execution of an instruction into several sub-steps which are executed by as many as hardware blocks which work like an "assembly-line". Each stage of the pipeline performs one (sub) operation in the execution. It get its data (the instruction in execution) in its input registers and put its results in its output registers. All these registers constitute an execution queue where instruction are progressively executed. The hardware complexity of a pipeline computer is several time larger than those necessary for sequential execution.



**Figure 12** Hardware structure of a pipe-line.

At each clock cycle a new instruction start execution by entering in the pipeline.



**Figure 13** Intructions in a pipe-line.

The main drawback with pipeline architecture is the dependency problem. They are two main cases of dependency:

- The first case of dependency is called data dependency. It occurs when an instruction in the pipeline wants to fetch data which has not yet been computed, because this data must be computed by another instruction deeper in the pipe.
- The second case of dependency is called instruction dependency. It occurs when the computer does not know the next instruction to be fetched because the condition used in a conditional branch instruction is not yet evaluated.

Obviously, these dependencies can be solved by stopping the problematic instruction (and the following ones!) in the pipe until the missing data or condition will be computed. The cost of this technique is very high because statistical measurements show that the probability of a dependency become high after 3 instructions.

## 15 VIRTUAL REGISTERS

One solution to eliminate data dependency is the use of virtual registers. In the registers of the pipeline, the places used to store the value of the operands of the instructions can be seen as virtual registers which can be directly accessed. When an instruction is waiting for a data not yet executed, the place used to store the value of this data is seen as a virtual register and the fetching of the data from memory is canceled. When the instruction which must compute this data has performed its computation the result is, in parallel, loaded in the different virtual registers of the instructions which are waiting for it, and also loaded at its place in memory. This short cut of the memory access and the elimination of many fetching of data from memory increases the performance of the computer.

## 16 BRANCH PREDICTION

Instruction dependency can be reduced by making some hypothesis on the behavior of the conditional branch. Better are the hypotheses, better is the performance improvement. When such an hypothesis becomes false, the instructions which were fetched in the pipe must be dropped. To allow such a back-tracking, the instructions which are fetched upon an hypothesis are marked as conditional. These instructions are not allowed to modify the programming context of the computer (i.e. writing in the programming registers and in memory).

Several techniques are used to predict branch instruction behavior. We will distinguish between the static and dynamic ones.

## 17 STATIC PREDICTION TECHNIQUES

The static prediction techniques are very simple:

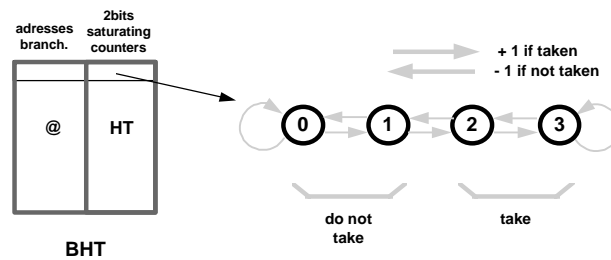
- The first one consists of not taking any conditional branch instruction. Its probability of success is about 40%. It was used in the 486 microprocessor.
- The second one consists of taking all branch instructions. Its probability of success is about 60%.
- An improvement of this technique consists of taking only the backward branch instructions, never the forward ones (it is called BTFN for Backward Take Forward Never). Its probability of success becomes 65%. It is used in the Micro-Sparc 2 and in the PA RISC 7x00.

- Another technique, which was used in the Ridge processor, consists in adding a bit to the branch instruction format. This bit is set by the compiler to indicate if the best hypothesis is to take or not the branch. The probability of success of this technique is about 75%.

## 18 DYNAMIC PREDICTION TECHNIQUES

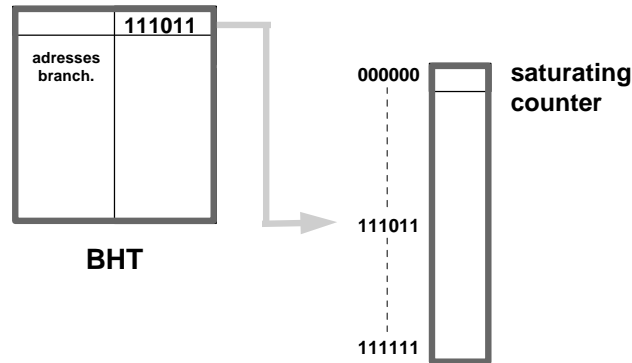
The dynamic prediction techniques are based on the measurement of the behavior of each branch instruction:

- The first technique developed by James Smith in 1981 consisted in recording the behavior of each branch instruction in order to predict the last one. Obviously, the size of the buffer used is limited and only the behavior of the last branch instruction can be stored. The probability of success of this technique is about 80%. This technique is used in the ALPHA and K5 microprocessors.
- An evolution of this technique called BHT (for Branch History Table) used a saturating counter as an inertial element to measure the behavior of each branch instruction. The probability of success become 85%. This technique is used in many microprocessors: PPC 604-620, ALPHA, Nx586, Ultra Sparc, M1, Pentium.



**Figure 14** Branch History Table.

- A two level dynamical mechanism has been proposed by Yeh and Patt in 1991. This technique consists in recording the successive behaviors (behavior profile) of each branch instruction. In addition, a saturating counter is used to predict the future behavior for each recorded profile. The probability of success becomes very high ( 95% ). This technique is used in the Pentium Pro microprocessor



**Figure 15** Two level dynamical mechanism.

## 19 OTHER COMPLEMENTARY TECHNIQUES

Because branch prediction is very important in term of performances, some complementary techniques are also used:

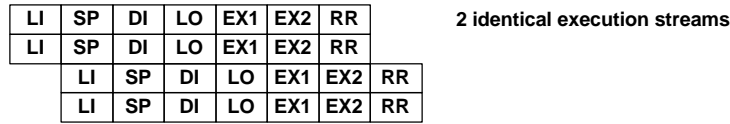
- Recording the addresses of computed branch in an associative memory
- Use of an hardware stack to hold the return address of the subroutines and interrupt (in addition of the classical system-stack located in memory). This technique is used in the M1 and ALPHA microprocessors.
- Use of an alternative cache called BTC (for Branch Taken Cache) for pre-decoding the first instructions of each alternative way.

## 20 SUPER-PIPELINE

The efficiency of the techniques to reduce the drawback of the dependencies have lead the designers to increase the depth of pipeline in order to decrease the amount of job performed between each stage. Such an architectural evolution allows to decrease the clock cycle of the computer. With this technique, the rate of the instructions entering in the pipeline is increased. The computer starts more instruction per second and its performance is increased (if the dependency problem is not increased!). As example the MIPS R 4000 microprocessor use a 8 step pipeline.

## 21 SUPER-SCALAR

Another acceleration technique consists in putting several pipelines in parallel in order to increase the throughput of the computer. These pipelines can be specialized or identical. Some computers use a specific pipeline for integer computation and another for floating point. In this case each pipeline fetches only the instruction it can execute. The specialized pipelines use different set of resources (registers). They are no dependency between them. When the pipelines are identical they share many resources (registers). Many conflicts and dependencies can occur. In this case it is necessary to extend to all the pipelines the notion of virtual registers.

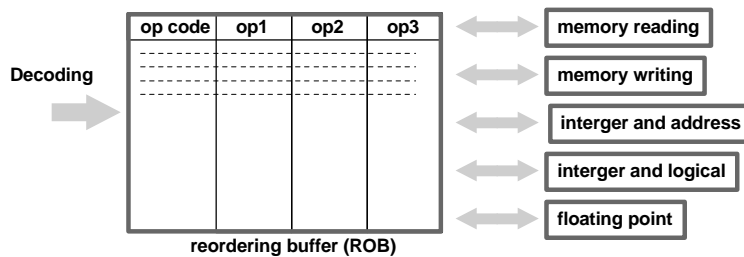


**Figure 16** The dual pipeline of the ALPHA microprocessor.

As example the ALPHA microprocessor uses two identical pipelines.

## 22 OUT-OF-ORDER EXECUTION

The most powerful technique to speed-up execution is the out-of-order execution technique which consist of executing an instruction as soon as possible without waiting for the completion of the previous ones. This kind of execution technique is organized around a specific buffer called ROB (for ReOrdering Buffer). In this buffer the instructions are progressively executed ("build") by specific operators. A fetching operator puts instructions in the ROB, detects data dependencies and removes them by using the technique of virtual registers. In the ROB the instruction format is far larger than the original one. It contains specific fields to hold the value of the operands and result. All the other operators are independently looking in the ROB to find instructions they can make progress. As example the operator for data fetching is looking for the instructions which need operands from memory. This operator fetches requested data from the memory and puts them into the value field of these instructions in the ROB. They are several operators for execution (integer, floating point, character string and addresses computation,...). Each of them is looking for the instructions which have all the values of their operands in the ROB. When it finds one, it computes its results and puts them in the ROB. This operator puts its result in the result field of these instructions but also in the virtual registers waiting for this data. The store operator looks for the instructions which have their result stored in the ROB. When it finds one, it stores the content of its result field in the memory and removes this instruction from the ROB.



**Figure 17** Out-of-order execution mechamism.

This very powerful (and costly!) technique is used in the Pentium Pro microprocessor.

## 23 CISC TO RISC TRANSLATION

In order to benefit of the advantages of the RISC architecture, the designers of CISC microprocessors have implemented an execution technique consisting of splitting each fetched instruction into a sequence of RISC instructions. This sequence of RISC instructions is put in the ROB and is executed by an out-of-order technique. As an example the Pentium Pro microprocessor uses this technique.

#### 24 USE OF VERY LONG INSTRUCTION WORDS (VILW)

In VLIW computers, several instructions (having no dependencies) are packed together in long words ended by branch instructions. These packed instructions are executed in parallel by several hardware streams. In these computers the long words are prepared by the compilers. In classical computers it seems possible to use this technique by designing a specific module which dynamicaly extract such long words from the cache to feed several execution streams in parallel.

#### 25 IMPROVMENT OF THE MEMORY ACCESSES

The access time of memories do not decrease as fast as the cycle time of the processors.

RISC microprocessors solve this problem by using many registers decreasing the need of memory access.

The CISC microprocessors use intensively the memory and are more difficult to optimize.

Memory (and registers) are single resources which produce bottleneck. The use of virtual registers decreases this problem.

#### 26 MEMORY HIERARCHY

All the new microprocessors use several levels of caching to accelerate the memory access. All of them have one or two inboard caches in the chip. Putting a cache on the chip puts it in the same microscopic world as the processor. The communication between the processor and its cache become faster and the number of connections between them can be large. Many microprocessors have two separated inboard caches. One for the data and the other for the instructions. The strategies of these cache can be separately optimized for the data and for the instructions.

Because the inboard caches are relatively small their efficiency in term of hit ratio is not very high. A secondary level of (large) cache is necessary. It is located on the motherboard or on the same chip carrier of the processor (e.g. Pentium Pro). Probably the evolution of microprocessors will lead to an increase in the size of the inboard caches.

#### 27 MULTIPROCESSOR IN A CHIP

Because microprocessors have already used all advanced techniques to speed-up execution, several authors suggest to develop multiprocessors on single chips. Such an idea is not new and several projects have already been investigated (As example a project of double MC 6800 has been investigated at the end of 70's). Unfortunately for these projects, each time the next generation of single processors was offering better advantages, the project was abandoned. Now

the situation may have slightly changed because all the known advanced execution techniques have already been used (except for the large inboard caches).

It is possible to imagine twin processors in a chip. The popular operating systems like: UNIX, Window NT, ... are already able to use them efficiently. Another solution is to specialize one processor for the input/output and visualization and the other for the user programs.

The percentage of the use of each processor will be low, but this is not really important. The development of personal workstations and PC is based on the low use of the hardware resources. As example a mainframe system is organized and tuned to maximize the use of its hardware resources. In a PC the hardware resources are poorly used, but the global economical result is more interesting.

## 28 CONCLUSIONS

Microprocessor evolution is one of the most exciting technical story of the end of this century. The computing power of these devices has been multiplied by a factor of several thousand during the last 25 years. No other technical domains have had such an evolution.

The microprocessors are progressively substituting themselves to the other technologies of processors. Probably they will absorb the range of supercomputer at the beginning of the next century.

To-day, the processors can be seen as a set of several sub-processors working for a single program. Possibly in the future, the microprocessor chip will hold several processors working in parallel for several program threads.

We can ask where such an evolution leads us. The planed computing power is very impressive and will be given to anyone to use the best techniques for simulation, CAD, virtual worlds, etc... making personal computers the best tools we have.

## 29 BIBLIOGRAPHY

- Anceau, F. (1986) The Architecture of Microprocessors. Addison-Wesley.  
Tredennick, N. (1986) Microprocessor Logic Design, Digital Press.  
Hennessy, J.L. and Patterson, D. A. (1990) Computer Architecture a Quantitative Approach, Morgan Kaufmann.  
Richard Comerford (1992) How DEC developed Alpha, IEEE Spectrum, July 1992  
Milutinovic, V. (1996) Surviving the Design of a 200 Mhz RISC Microprocessor, Lessons Learned. Computer Society.

Many information about microprocessor architecture are available in the Microprocessor Report newspaper and in the proceedings of the ISSCC Conference.

## 30 BIOGRAPHY

Prof. Anceau received the engineering degree from the Institut National Polytechnique de Grenoble (INPG) in 1967, and the Docteur d'Etat degree in 1974 from the University of Grenoble. He started his research activity in 1967 as member of the Comite National pour la Recherche Scientifique (CNRS) in Grenoble. He became Professor at INPG in 1974 where he led a research team on microprocessor architecture and VLSI design. In 1984, he moved to industry (BULL company, close to Paris) to lead a research team on Formal Verification for

Hardware and Software. The first industrial tool for hardware formal verification and the technique of symbolic state traversal for finite state machines was developed in this group. In 1996 he took his present position as Professor at Conservatoire National des Arts et Metiers (CNAM) in Paris. Since 1991 he has also been a Professor at Ecole Polytechnique in Palaiseau, France. His main domains of interest are: microprocessor architecture, VLSI design, hardware formal verification, and research management and strategy. He has given many talks on these subjects. He is the author of many conference papers and of a book entitled "The Architecture of Microprocessors" published by Addison-Wesley in 1986. He launched the French Multi-Chip-Project, called CMP, in 1981.