**Techninis ir istorinis kontekstas:**

Zilog Z80000 yra neišleistas 32 bitų procesorius pagamintas 1986 metais. Ši architektūra buvo pagaminta praplėčiant 16 bitų Zilog Z8000, tačiau negali vykdyti kodo parašyto Z8000 arba Z80 procesoriams. Procesorius daugeliu atžvilgių yra panašus į „Intel“ 80386. Dėl pradinės gamybos vėlavimo Z80000 pasiekė tik bandomasis testavimo etapas, niekada nebuvo nepaleistas komerciškai.

PA-RISC (Precision Architecture [RISC](https://en.wikipedia.org/wiki/Reduced_instruction_set_computer)) architektūra buvo pagaminta [Hewlett-Packard](https://en.wikipedia.org/wiki/Hewlett-Packard) kompanijos 1986 metais. Pagal šią architektūrą pagamintos sistemos buvo pardavinėjamos iki 2008 metų, tačiau iki 2013 metų palaikė serverius, kuriuose veikia PA-RISC lustai. Pagal PA-RISC architektūrą buvo sukurtas Intel Itanium procesorius.

**Kompiuterio/procesoriaus bazė:**

Z80000

* **Technology:** Integrated Circuits (ICs)
* **Integration Scale:** **Large Scale Integration (LSI)**. This refers to the integration of thousands of transistors on a single chip.
* **Transistors:** It used **metal-oxide-semiconductor (MOS)** technology, which was common in the late 1970s and early 1980s.

PA-RISC

**Architektūra:**

Z80000 procesorius turi šešiolika 32 bitų registrų. Komandų sistemoje yra 9 adresavimo būdai, operacijos su bitais, bitų laukais, baitais, žodžiais (16 bitų), ilgais žodžiais (32 bitų) ir kintančio ilgio tekstinėmis eilutėmis (iki 65,536 baitų) . The memory management, exception handling, and system and normal mode features support the development of reliable software systems.

PA-RISC

**Kokių adresų mašinos:**

Z80000

PA-RISC architektūra yra trijų adresų mašina – įprastai perskaito du operandus iš bendro naudojimo registrų, atlikus operacijas rezultatas yra padedamas atgal į bendro naudojimo registrą.

**Registrai:**

Z80000 procesorius turėjo šešiolika 32 bitų fizinių registrų, kurie buvo pagaminti praplėčiant 16 bitų Z8000 procesorių. Šioje architektūroje buvo galima pasiekti ir 16 bitų, ir 8 bitų registrus, esančius viduje 32 bitų registrų, taip pat galima buvo sujungti du registrus ir tada procesorius interpretuodavo juos kaip aštuonis 64 bitų registrus.

Bendro naudojimo registrai talpina 64 baitus atminties.

* Pirmi 16 baitų (RL0, RH0, ..., RL7, RH7) naudojami kaip akumuliatoriai baitų ilgio duomenims.
* Pirmi 16 žodžių registrai (R0, R1, ..., R15) naudojami kaip akumuliatoriai žodžių ilgio duomenims, kaip indeksavimo registrai (išskyrus R0), arba atminties adresams kompaktiškame režime.
* Bet kuris ilgo žodžio registras (RR0, RR2, ..., RR30) gali būti naudojamas skaičiavimas su ilgais žožiais, arba kaip indeksų registras (išskyrus R0) segmentavimo ar tiesiniame režime, arba atminties adresams.
* 64 bitų registrai (RQ0,RQ4, ..., RQ28) gali būti naudojami kaip akumuliatoriai dauginimui, dalinimui ir ženklo praplėtimui.
* Within quadword register RQn, RRn contains the more significant longword. A 4-bit field in instructions specifies which general purpose register to access. The register size is determined by the instruction opcode.

PA-RISC turėjo 32 32-bitų bendros paskirties registrus.

* Nulinis registrass visada gražina nulį, ir kai į jį rašome, reikšmė vistiek išlieka nulis.
* 25 valdymo registrus, kurie yra naudojami virtualių adresųų apsaugojimui, pertraukimų apdorojimui ir įvairioms funkcijoms.
* 8 erdvės registrai naudojami sukurti virtualiems adresams.

**Požymių bitai:**

Z80000 architektūroje buvo naudojami šeši požymių bitai:

* pernešimo (C)
* nulio (Z)
* ženklo (S)
* lyginumo/perpildymo (P/V)
* dešimtainių skaičių ... (Decimal Adjust (D)) - yra naudojamas dvejetainių-dešimtainių skaičių (BCD binary coded decimal) skaičiavimuose, kad būtų sekama, koks paskutinis veiksmas buvo atliktas – sudėtis ar atimtis
* pusinio pernešimo (H) – taip pat yra naudojamas yra naudojamas dvejetainių-dešimtainių skaičių (BCD) skaičiavimuose, kad paversti rezultatą iš buvusios atimties ar sudėties į dešimtainį skaičių.

PA-RISC architektūroje išskirtų požymių bitų neradau, tačiau ši architektūra turi procesoriaus būsenos žodį (Processor Status Word PSW), kuriame yra įvairūs sistemos valdymo bitai, prioritetų magistralėse bitai, tačiau juose taip pat yra 8 pernešimo/pasiskolinimo bitai, naudojami atliekant aritmetinius veiksmus.

**Mašininis žodis**

Abiejų architektūrų mašininio žodžio plotis buvo 32 bitų.

**Atminties išdėstymas, adresų erdvė, efektyvus adreso plotis, maksimalus įmanomas atminties kiekis, tipiškas atminties kiekis**

Z80000

Z80000 procesorius turėjo tris adresavimo būdus: (nuo 20psl jei truks info)

* Compact – naudojamas tų programų, kurių adresų erdvė yra mažesnė nei 64Kb ir gali efektyviai naudoti 16-bitų adresus
* Segmented – palaiko 2 segmentavimo dydžius – 64Kb ir 16Mb. In segmented mode, address calculations do not affect the segment number, only the offset wi thin the segment. 32,768 segments of 64 KB (16-bit address; comprising memory from 0-2GB) *and* 128 segments of 16 MB (24-bit address; comprising memory from 2GB-4GB), making a total of 4 GB (32-bit address) of accessible memory.
* Linear - The 32-bit addresses in linear mode provide uni form and unstructured access to 4G bytes of memory. Some applications benefit from the flexibil ity of linear addressing by allocating objects to arbitrary positions in the address space

Kurie buvo parenkami pagal valdymo bitus požymių bitų ir valdymo žodžio registruose.

PA-RISC HP Precision processors access memory using byte ad dresses. Larger addressable units include half words, words, and double words. An address is either physical or virtual. All load and store instructions can be used in either virtual or physical mode. Virtual mode is enabled sepa rately for instruction fetches and data accesses by two flags in the processor status word. A pointer to physical memory is a 32-bit unsigned integer whose value is the address of the first byte of the operand it designates. Physical addresses are used directly, with no protection or access rights checking performed. Virtual ad dresses are translated to physical addresses and undergo protection and access rights checking as part of the trans lation. This allows the hardware support for access control to be built into the storage unit.

**Virtuali atmintis:** Z80000 architektūroje virtualioji atmintis buvo palaikoma puslapiavimu.

Procesorius įgyvendina panašų puslapiavimo mechanizmą kaip ir kitose architektūrose.Operacinė sistema sukuria virtualių adresų lenteles atmintyje, tada užkrauna rodykles į tas lentelės valdymo registruose. Procesorius automatiškai kreipiasi į lenteles atimintyje, kad perkeltų virtualios atminties adresus į fizinę atmintį ir patikrintų, ar galima tą atminties dalį pasiekti. Kad suvaldyti didelę adresų erdvę, adresų vertimo sistema padalina atmintį į fiksuoto dydžio 1Kb puslapius.

PA-RISC architektūroje yra trys lygiai, jie priklauso nuo to, kiek virtualios atminties yra palaikoma.

0-nis lygis nepalaiko jokios virtualios atminties. Pirmo lygio procesorius plaaiko 16-bitų erdvės registrus skirtus 48 bitų virtualios atminties erdvei ir trečio lygio procesorius įgyvendina 32-bitų erdvės registrus kur palaikoma pilna 64 bitų virtuali adresų erdvė.

Virtualus adresas yra apibrėžtas globaliai ir yra toks pat, nepriklausomai nuo to, koks procesas jį naudoja. Kitose architektūrose dažnai tas pats virtualus adresas gali būti naudojamas pasiekti skirtingus objektus skirtingiems procesams, kaip yra ir Z80000 architektūroje. PA-RISC architektūroje virtuali atmintis taip pat palaikoma puslapiavimu – vieno puslapio dydis 2Kb

The virtual address space is so large that processes can be assigned separate address ranges for private data. Address translation information does not need to change upon a process switch and the information needed for address translation can be represented more compactly. Global virtual addressing therefore allows closely coupled processes to accumulate a stable working set of address translations in spite of fre quent process switching.

Virtual memory is structured as a set of address spaces, each containing 2^32 bytes. A level-one processor imple ments 2^16 spaces (16-bit space registers), and a level-two processor implements 2^32 spaces (32-bit space registers). A space is specified by a space identifier, and is divided into pages, each 2048 bytes in length. For a level-two processor, the concatenation of a 32-bit space identifier and a 32-bit offset within the space forms a virtual address. Alternatively, a virtual address may be viewed as the concatenation of a 53-bit virtual page number and an 11 -bit offset within the page. For virtual addressing, space identifiers are specified in space addressing registers. These include the space portion of the instruction address register and the eight space regis ters SR 0 through SR 7 (see Fig. 4). One such register is implicitly or explicitly selected by every instruction that generates a virtual address. (jei truks info tai 14 psl)

**Komandų sistema:**

Z80000 architektūros komandų sistema turėjo apie 800 skirtingų komandų. Turėjo 11 instrukcijų klasių:

* Įkėlimo ir išsaugojimo (EX dst, src – Exchange)
* Aritmetinės (ADC dst, src – Add with Carry; CP dst, src – Compare)
* Loginės (XOR dst, src – Exclusive or)
* Programos valdymo (JP cc, dst – Jump)
* Veiksmų su bitais (SET dst, src – Set bit)
* Sukimas, perslinkimas (RCL dst, src – rotate left through carry)
* Blokų perkėlimas ir komandos su tekstinėmis eilutėmis (CPSIR dst, src, r, cc – compare string, increment and repeat)
* Įvesties ir išvesties (OTIR dsr, src, r – output, increment and repeat)
* Procesoriaus valdymo (DI int – disable interrupt; EI int – enable interrupt)
* Papildomos instrukcijos – jos yra skirtos papildomiems procesoriams, pavyzdžiui Z8070 ALU palaiko slankaus kablelio skaičių aritmetiką.

PA-RISC turėjo 140 komandų. Jos yra išskirstytos į šias klases:

* FLOP – operacijos su slankaus kablelio skaičiais
* LDST – komandos darbui su atmintim – įkėlimas ir išsaugojimas
* ALU – veiksmai su sveikais skaičiais
* MM – poslinkiai, išskyrimai, įterpimai
* NUL – galintys anuliuoti kitą veiksmą
* BV – vietinė vektorinė šakos, išorinės šakos
* BR – kitos šakos
* FSYS – FTEST ir slankiojo kablelio būsena/įvykiai
* SYS – sistemos valdymo instrukcijos

In HP Precision Architecture, all instructions have a fixed length of thirty-two bits, which is one word of memory. Time-critical functions are placed in fixed-position fields, so that they can proceed with minimal or no decoding. Since all instructions are word-aligned, an instruction never crosses a page boundary.

Kokius instrukcijų formatus palaikė kiekviena architektūra? Kokios komandos buvo panašios abi architektūros? Kurios komandos skyrėsi?

**Adresavimo būdai:**

Z80000 procesorius dirba su šiais operandais – registrais, atmintims, periferiniais prievadais, ar tiesioginiais operandais esančiais komandoje. Jiem pasiekti ir su jais dirbti naudojami 9 adresavimo būdai: registrinis (Register), tiesioginių operandų (Immediate), netiesioginio registro (Indirect Register), tiesioginės prieigos (Direct Address), indeksavimo (Index), bazinio adreso (Base Address), bazinio indekso (Base Index), asociatyvaus adreso (Relative Address), and asociatyvaus indekso (Relative Index).

PA-RISC turėjo tik tris adresavimo būdus - long/short displacement and indexed

**I/O galimybės**

Z80000 Although logical I/O addresses are 32 bits, only the 16 low-order bits of a logical I/O address can be manipulated; the CPU always forces the 16 high order bits to O. Unlike logical memory address spaces, logical I/O address space is not viewed as a string of bytes at consecutive addresses. Rather, the address is simply used to locate a byte, word, or longword peripheral port. The byte port located at address n does not have to be contiguous with the byte port located at address n+1, nor must it be the more significant byte of the word port located at address n. Logical I/O addresses can be either even or odd.

PA-RISC

Įvesties/išvesties (I/O) architektūroje atmintis yra susieta. Visas I/O sistemos kontroliavimas yra vykdomas su Įkėlimo ir išsaugojimo (load and store) komandomis dirbant su virtualiais ar fiziniais adresais. Tai leidžia I/O programoms būti rašomoms aukšto lygio kalbomis.

Furthermore, since the usual page-level protection mechanism is applied during virtual-to-physical address translation, user programs can be granted direct control over particular I/O devices without compromising system integrity.

**Pertraukimai**

The Z80,000 palaiko tris pertraukimų rūšis:

* neužmaskuojamas (nonmaskable) -
* vektorinius (vectored)
* nevektorinius (nonvectored).

Neužmaskuojamas pertraukimas – aukščiausio prioriteto, yra rezervuojamas kritiniams atvejams, tokiem kaip staigus energijos nutrūkimas. Vektoriniai ir nevektoriniai pertraukimai gali būti paslepiami su požymių bitais valdymo žodžio registre. Vektoriniai pertraukimai leidžia iššakoti pertraukimą į atskirą išimčių tvarkyklę, parinktą kode. Ne vektoriniai pertraukimai naudoja paprastą pertraukimų tvarkyklę.

PA-RISC architektūra įgyvendina vieno lygio pertraukimų sistemą. Tai reiškia, kad kai yra vykdomas pertraukimas, jo negalima pakeisti kitu, nors jis ir būtų aukštenio prioriteto, ir tai reiškia, kad vienu betu gali būti vykdomas tik vienas pertraukiamas. Jeigu reikia kelių pertraukimų vienu metu – vykdoma pagal prioritetų eilę, ir kai pertraukimas įvykdomas, komanda vėl iškviečiama, kad būtų iškviesti kiti, žemesnės eilės pertraukimai. Pertraukimų įterpimas vienas į kitą yra leidžiamas, pertraukimų tvarkyklė gali vėl įjungti kitus pertraukimus, kai yra pasiekta reikiama būsena. Kai kuriuose konvejeriniuose procesoriuose pertraukimai nėra tikslūs, kartais jie nėra įvykdomi iškart po instrukcijos, kuri sukėlė tą pertraukimą.

**Duomenų tipai:** Z80000 procsorius palaikė komandas su 9 duomenų tipais: bitais, bitų laukais, sveikais skaičiais su ženklu, sveikais skaičiais be ženklo, loginėmis reikšmėmis, adresais, BCD (dvejetaine skaičiavimo sistema užkoduotas dešimtainis skaičius), stecke esančiomis reikšmėmis, tekstinėmis eilutėmis. Sveikas skaičius ir loginės reikšmės gali būti baito, žodžio, ar ilgo žodžio dydžio. Veiksmai su slankaus kablelio skaičiais yra įgyvendinami Extended Processing Architec ture (EPA) facility by a coproceasor (ZB07D Arith metic Processing Unit) or by software emulation. Yra galimybė atlikti veiksmus su trupmeniniais skaičiais remiantis Support is provided for Binary Coded Decimal (BCD) arith metic and multiple precision arithmetic.

PA-RISC

Sveiki skaičiai su ženklu ar be ženklo gali būti 8, 16 arba 32 bitų ilgio. Skaičiai su ženklu įgyvendinami papildinio iki dviejų logika abiejose architektūrose. Simboliai yra 8 bitų ilgio ir palaiko ASCII standartą. Slankaus kablelio skaičiai yra saugomi 32 arba 64 bitų ilgio, kurie palaiko ANSI/IEEE 754-1985 standartą. BCD irgi..

**Sistemos greitaveika:**

Z80000 The system designer can fine-tune performance by selecting not only the CPU clock rate and bus speed (1/2 or 1/4 the CPU clock), but also the access time and date path width for the memory. for two independent regions of memory the CPU can be programmed for both the number of wait states automatically inserted, and whether the data path is 16 or 32 bits wide. With these options, a system can essily accommodate slow, 16-bit-wide bootstrap read-only memory (ROM) in one region and fast, 32-bit-wide random access memory (RAM) in the other. furthermore, the CPU supports an optional burst transfer of several memory words from consecutive locations. Burst transfers can increase memory bandwidth for interleaved and "nibble-mode" memory systems.

PA-RISC - A primary design goal was that all functional computa tions in the basic instruction set could execute in one machine cycle in a pipelined implementation of the proces sor architecture. Operations were selected for inclusion in the basic instruction set only if they could be implemented in a reasonably small number of logic levels, to guarantee a short cycle time. This does not necessarily mean that the operation performed had to be primitive in function. In fact, rather sophisticated operations were allowed in the architecture if they proved useful to the compilers, and were implementable in a short machine cycle with rela tively simple hardware. ively simple hardware. Complex operations that are necessary to support re quired software functions but cannot be implemented in a single execution cycle are broken down into primitive op erations, each of which can be executed in a single cycle. Single-cycle execution was a design goal of the architec ture, but is not a constraint on the implementations. For example, an HP Precision microprocessor may operate with slower memories, performing a load instruction in more than one cycle.

This is the data flow for the basic three-register model of execution, which facilitates single-cycle execu tion, since no memory references are required.

**Spartinančioji atmintis**: Z80000 įgyvendina spartinančios atminties mechanizmą, kuri laiko kopiją neseniai naudotos atminties vietos/adreso vietą luste. Ten laikomos ir instrukcijos, ir duomenys. Kai bandoma pasiekti atmintį, procesorius patikrina, galbūt tai ko ieškome yra spartinančioje atmintyje. Jeigu neranda, perkopijuoja tą informaciją iš atminties į spartinančiąją atmintį vietoj seniausiai naudotų duomenų. Taip sumažinama atminties skaitymo operacijų. Spartinančioji atmintis yra 256 baitų dydžio.

PA-RISC

**Kur buvo naudojamos:**

The regular combination of addressing modes, operations, and data types offers a powerful instruction set that is well-suited for compila tion of high-level languages such as C, Pascal, and Ada. Z80000

**Kiek programinės įrangos parašyta:**

Z8000 COMPATIBILITY The 180,000 CPU's instruction set encoding allows it to directly execute Z8000 family software such as compilers and the 1RTS~ real-time operating system. Z8000 programs must not use the 18000 privileged instructions, address, and control field encodings if they are to execute correctly on the Z80,000 CPU, since the Z80,000 CPU uses many of these reserved encodings to extend the register file, address range, and instruction functionali t y.

The computers covered on this site, the [HP 9000](https://www.openpa.net/systems/), are based on the Precision Architecture and [PA-RISC processors](https://www.openpa.net/pa-risc_processors.html) and used custom HP system designs.

**Šaltiniai:**

<https://en.wikipedia.org/wiki/Zilog_Z80000>

<https://www.openpa.net/pa-risc_architecture.html>

<https://web.archive.org/web/20110712230227/http://www.hpl.hp.com/hpjournal/pdfs/IssuePDFs/1986-08.pdf>

<http://ftp.parisc-linux.org/docs/arch/pa11_acd.pdf>

<https://parisc.docs.kernel.org/en/latest/>

<https://bitsavers.trailing-edge.com/components/zilog/z80000/Z80000_CPU_Preliminary_Technical_Manual_Sep84.pdf>