

Technologie procesoru Intel 80386

Otázky

První část – Základní parametry

- Obecně
- Základní parametry
- Architektura
- Módy činnosti

Registry, adresace paměť

- Registry
- Adresace paměti
- Ukazatel
- Tabulky popisovačů objektů

Přerušení, obsluha výjimek

- Obecně
- Výjimky
- Obsluha přerušení
- Mechanismus obsluhy

První část - Základní parametry

Obecně

Procesor 80386 je první 32 bitový procesor od firmy Intel.

Základní parametry

Uvedení na trh: 1985

Hodinová frekvence: 16 – 33 MHz

Výkon: 6 - 15 MIPS (instrukcí za sekundu)

Vnitřní datová sběrnice: 32 bitová

Vnější datová sběrnice: 32 bitová

Adresová sběrnice: 32 bitová → adresace až 4GB paměti (64TB virtuální)

Napájecí napětí: +5V

Počet tranzistorů: 275 000

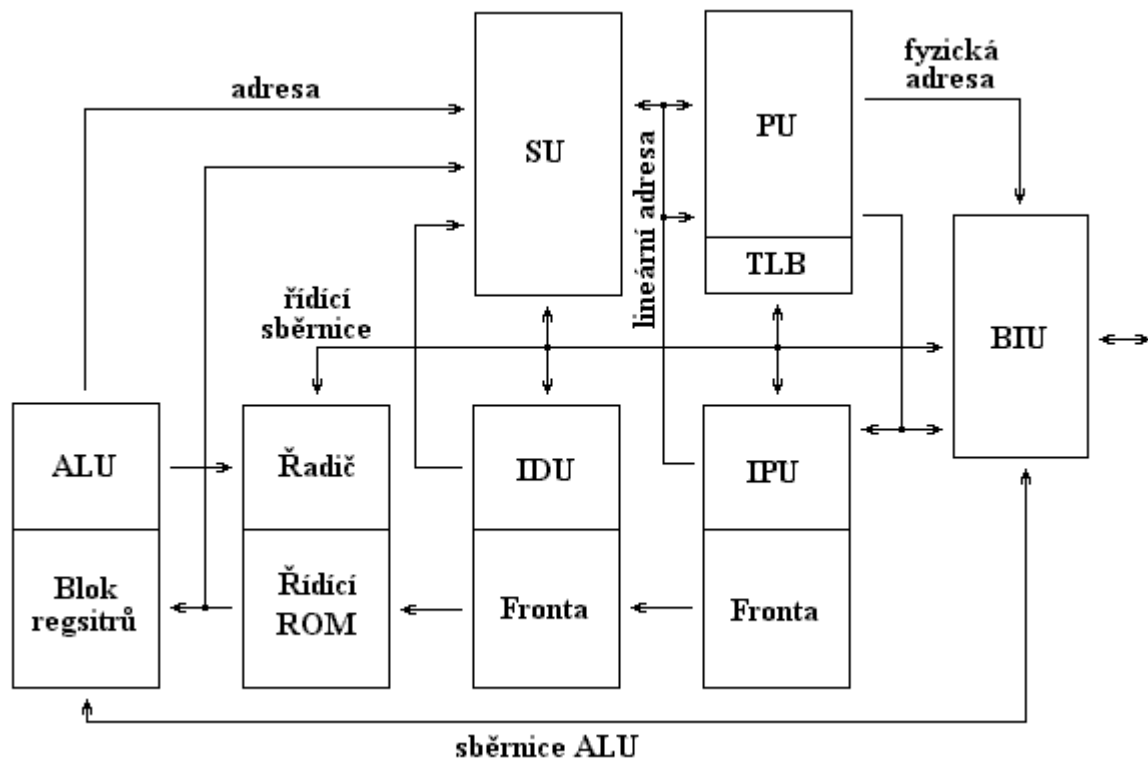
Pouzdro: PGA, 132 vývodů, keramické

Architektura: CISC

Architektura

Procesor je složen ze 6 nezávislých paralelně pracujících jednotek:

- **BIU** (Bus Interface Unit - jednotka styku se sběrnici): Tato jednotka je branou mikroprocesoru k okolnímu světu. Všechny ostatní jednotky procesoru využívají tuto jednotku pro přenos dat mezi procesorem a okolím. Protože BIU pracuje výhradně s fyzickými adresami, je nutné, aby adresa, která je poskytována k provedení operace, byla nejprve převedena na fyzickou adresu.
- **IPU** (Instruction Prefetch Unit - jednotka předvýběru instrukcí): Tato jednotka se stará o naplňování šestnáctibytové fronty předvybraných instrukcí pro IDU. IPU nepřetržitě požaduje po vyzvednutí instrukce z fronty IDU, aby BIU doplnila frontu z následující adresy. Instrukce jsou do fronty zapisovány po 4 bytech. V případě provedení instrukce, která způsobí skok, provede IPU vyprázdnění celé fronty a další plnění se provádí od nové adresy.
- **IDU** (Instruction Decode Unit - jednotka pro dekodování instrukcí): Jednotka, která má podobnou funkci jako IPU. Vyzvedne z fronty naplněné IPU první byte instrukce a podle něj zjistí délku celé instrukce (může být dlouhá až 16 B). Pak vyzvedne z fronty celou instrukci (popř. požádá BIU o doplnění chybějící části) a převede ji na vnitřní formát. Takto dekodovanou instrukci umístí do své fronty dekodovaných instrukcí, která je schopna pojmout až 3 dekodované instrukce. Zde je instrukce uložena pro potřebu EU.
- **EU** (Execution Unit - prováděcí jednotka): Jednotka, která provádí vlastní výpočty. Jejím jádrem je **ALU** (Arithmetic - Logic Unit), která obsahuje obvody potřebné k aritmetickým a logickým operacím a k provádění instrukcí. Obsahuje také sadu registrů procesoru. Posledním úkolem EU je informovat BIU, že výsledek je potřeba zapsat do operační paměti nebo na periferní zařízení.
- **SU** (Segmentation Unit - jednotka segmentace): Tato jednotka má význam především v chráněném a virtuálním režimu, kdy provádí převod virtuální (logické) adresy na adresu lineární.
- **PU** (Paging Unit - stránkovací jednotka): Jednotka, která se uplatňuje pouze v chráněném a virtuálním režimu, a to jenom při zapnutém režimu stránkování. Potom PU provádí převod lineární adresy dané SU na adresu fyzickou. Ke své činnosti využívá rychlou vyrovnávací paměť TLB.



Módy činnosti

- **reálný režim**: plně kompatibilní s procesorem I8086. Procesor simuluje I8086.
- **chráněný režim (protected mode)** - nativní formát plně slučitelný s procesorem I80286. Dnes nepoužívanější.
- **virtuální režim** - v tomto režimu procesor 80386 pracuje podobně jako procesor 8086 (8088), ale je plně podřízen režimu chráněnému. Je možné takto virtualizovat 1 MB operační paměti, který mohl adresovat procesor 8086 a uložit jej kamkoliv do 4 GB operační paměti. Umožňuje v rámci chráněného módu spouštět programy určené původně pro procesor 8086 bez rekompilace. (Dosovské okno v systému Windows)

Registry, adresace paměti

Registry

Univerzální - univerzální registry může programátor využít jakkoli.

- AX - akumulátor (řada instrukcí ho má jako implicitní operand)
- BX - bazový registr (tj. dá se využít pro adresaci)
- CX - čítač (tj. určený pro počítání cyklů)
- DX - rozšíření akumulátoru

Indexové - Tyto registry slouží primárně pro adresaci v paměti

- ESI - source index - index pro zdroj (tj. pro čtení)
- EDI - destination index - index pro cíl (tj. pro zápis)
- EBP - určen jako ukazatel na záznam aktivní procedury na zásobníku (tím, že se implicitně spojoval s SS)
- ESP - ukazatel vrcholu zásobníku
- EIP - ukazatel kódu následující instrukce. Nelze k němu přímo přistupovat (jen pomocí instrukcí skoků)

Segmentové - Slouží k ukládání adresy segmentu – pomocí nich se adresuje paměť. Jejich viditelná (přístupná) část je pouze 16bitová.

- CS - segment kódu
- DS - datový segment
- ES - extra segment
- SS - zásobníkový (stack) segment
- FS a GS

Registři příznaků - Ukládají se do něj informace o stavu procesoru, úspěšnosti provedených instrukcí, atd.

- ZF (Zero Flag) – Příznak se nastaví, je-li výsledek operace nulový.
- DF (Direction Flag) – Určuje směr zpracování řetězců. Je-li nastaven v jedničce, obsah registrů SI a DI se po provedení řetězcové operace sníží (řetězec se zpracovává odzadu).
- PF (Parity Flag) – Příznak se nastaví, jestliže dolních 8 bitů výsledku prováděné instrukce má sudý počet jedniček (neznámá, že číslo je sudé).
- IF (Interrupt Flag) – Povoluje externí přerušení.
- CF (Carry Flag) – Příznak se nastaví, jestliže při provádění operace došlo k přenosu z nejvyššího bitu. Používá se také při instrukcích posunu a rotace.
- AF (Auxiliary Flag) – Příznak se používá při BCD v aritmetice a to při přenosu z 3. do 4. bitu.
- SF (Sign Flag) – Je-li výsledek operace záporný, příznak se nastaví do jedničky. Je to v podstatě kopie nejvyššího bitu výsledku. (příznak znaménka +, -)
- OF (Overflow Flag) – Příznak se zastaví do jedničky, došlo-li v průběhu provádění instrukce k aritmetickému přeplnění, tzn., že výsledek se nevejde do určeného místa.
- TF (Trap Flag) – Tento příznak se používá pro účely ladění, jeho nastavení způsobí, že procesor po provedení každé instrukce vygeneruje přerušení INT1.

Další registry:

Skupina registrů systémových adres – GDTR, LDR, IDTR, TR – báze rozšířena na 32 bitů

Řídící registry – CR0, CR1, CR2, CR3 – 32 bitové

Ladící registry – DR0, DR1, DR2, DR3 – uložení lineárních adres ladících bodů

Stavový registr – DR6

Příkazový registr – DR7

Pro testování a ladění mechanismu stránkování – TR6, TR7 – 32 bitové

Adresace

Reálný mód – Fyzická adresa = segmentový registr * 16 (posunutí o 4 řády doleva) + offset

Chráněný mód – Hodnota segmentového registru je ukazatel do tabulky popisovačů

Tabulka popisovačů obsahuje:

- Adresu počátku bloku paměti (segment) – 32 bitů
- Limit (maximální velikost segmentu) – 20 bitů
- Vlastnosti – 12 bitů

Formát ukazatele (obsah viditelné části segmentového registru)

Bit 0 – 1: Úroveň oprávnění (RPL)

- Úroveň 0 – Jádro OS, řízení CPU
- Úroveň 1 – Služby OS: přidělování prostředků, plánování úloh
- Úroveň 2 – Systémové programy z knihoven, souborový systém
- Úroveň 3 – Uživatelské aplikace

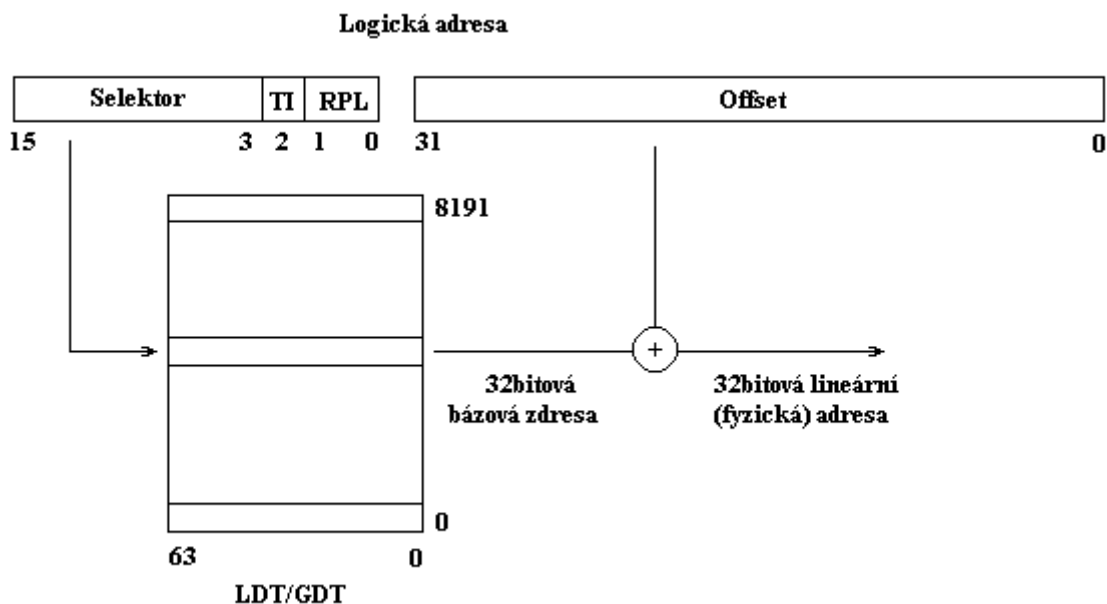
Bit 2: Volba tabulky popisovačů (TI)

Bit 3 – 15: Index do tabulky popisovačů (selektor)

Tabulky popisovačů objektů

2 třídy:

- **Systém**
 - **Tabulka popisovačů pro přerušení (IDT)** – Začátek tabulky a její limit udává systémový registr **IDTR**
- **Segment**
 - **Globální tabulka popisovačů objektů (GDT)** – Počátek tabulky a její limit v paměti je dán obsahem systémového registru **GDTR**. Jsou v umístěny objekty mající všeobecnou platnost
 - **Lokální tabulka popisovačů objektů (LDT)** – Počátek tabulky a její limit v paměti je dán obsahem systémového registru **LDTR**. Jsou v umístěny objekty mající místní platnost



Stránkování

Rozdíl mezi segmentováním a stránkováním je jiný způsob práce s pamětí. Segmentováním se programy a data dělí na různě velké segmenty. Při stránkování se ale paměť dělí na stránky pevné velikosti. Mechanismus stránkování funguje tak, že celá virtuální paměť (část pevného disku), kterou proces používá, je rozdělena na stejně velké oblasti. Ve fyzické paměti RAM je vyhrazena oblast, obsahující určitý počet stejně velkých oblastí – rámců. Platí pravidlo, že do jednoho rámu lze umístit jen jednu stránku. Pokud program pracuje a potřebuje data z virtuální paměti, přesunou se příslušné stránky do rámců. Stránkovací jednotka přepočítává lineární adresu na číslo rámu, ve kterém je teď stránka uložena. Jedna stránka má velikost 4kB a tudíž jeden rám má stejnou velikost.

32bitová lineární adresa je rozdělena na tři části:

- **Adresář:** tvořen nejvyššími 10 bity, slouží jako index tabulky zvané adresář, odkud je vybrána 20bitová báze tabulky stránek.
- **Tabulka:** tvořena nižšími 10 bity, slouží jako index do tabulky stránek. Z této tabulky se vyzvedne 20 bitová báze stránky, která tvoří 20 nejvyšších bitů fyzické adresy.
- **Offset:** tvořen nejnižšími 12 bity. Offset je potom sečten s o 12 bitů posunutou (vynásobenou 4096) bází stránky tak, že tvoří nejnižších 12 bitů fyzické adresy.

Přerušení, obsluha výjimek

Obecně

- **Přerušení** = Žádost o obsluhu, generovanou jak HW tak SW
= Asynchronní událost, jež vyžaduje neodkladnou obsluhu
- **Výjimka** = stav, který nastává, detekuje-li procesor nějaký chybový stav

Výjimky

1. **Fault** – Lze lokalizovat, mohou být ošetřeny, vykonávání může pokračovat. Řízení vráceno za instrukci, která stav způsobila. Např. přetečení.
2. **Trap** – Lze lokalizovat, mohou být ošetřeny, vykonávání může pokračovat. Řízení vráceno před instrukci, která stav způsobila. Např. dělení nulou.
3. **Abort** – Fatální. Nelze jednoznačně identifikovat. Úloha je ukončena

Obsluha přerušení

v chráněném módu:

a) procedurou

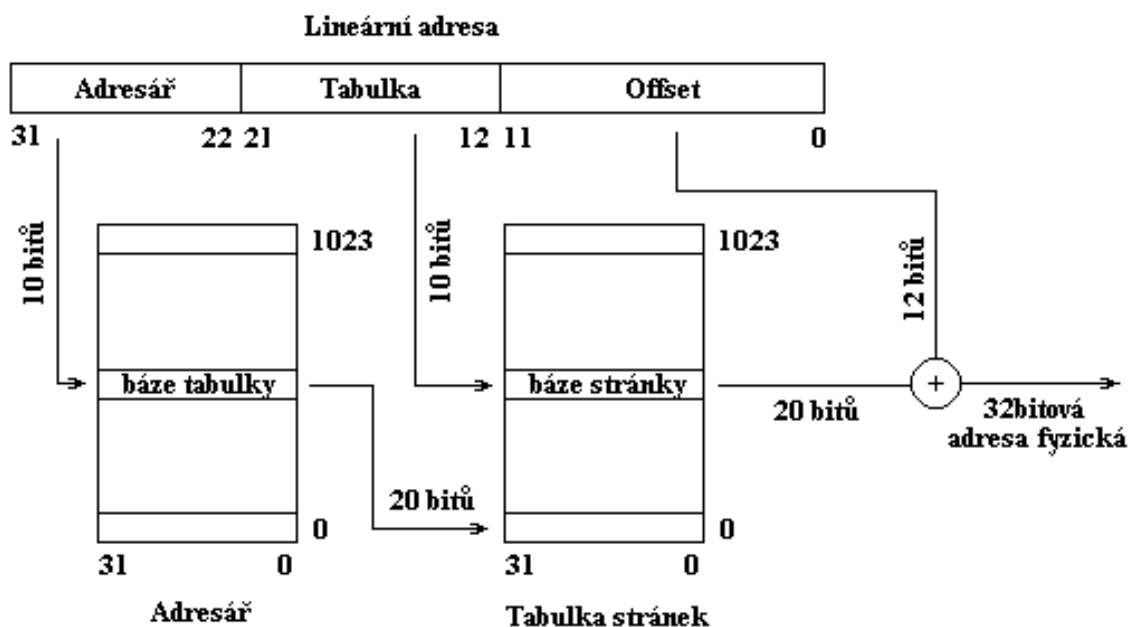
- jednodušší
- do zásobníku se uloží registr příznaků, návratová adresa a řízení se předá segmentu obsluhy přerušení

b) úlohou

- probíhá stejně jako kterákoli jiná úloha v systému
- vykonávání stávající úlohy je pozastaveno (suspend, uložení kontextu) a dojde k přepnutí úlohy obsluhy příslušného přerušení nebo výjimky

Mechanismus obsluhy

Registr IDTR ukazuje na tabulku popisovačů přerušení IDT, tady jsou potom pouze objekty typu systém.



Ilustrace 1: Stránkování