NVIDIA GeForce3

GEFORCE3
SNAD NE PŘEHNANÉ OČEKÁVÁNÍ

Už jsme si zvykli na to, že grafické čipy GeForce firmy NVIDIA jsou vždy o kousek napřed před konkurencí – jak v použité technologii, tak i ve výkonu. Je tomu tak i v případě nového čipu GeForce3? Na tuto otázku se budeme snažit najít odpověď v tomto článku.

NVIDIA rozhodně svůj trůn na poli herního grafického hardwaru nemíní opustit. Je pravda, že v některých oblastech jí začal trochu docházet dech, ale čipem GeForce3 jasně naznačuje, že se s druhým místem rozhodně nemíní spokojit. Oněmi problémovými oblastmi byly celoobrazovkový antialiasing a hlavně komunikace mezi čipem a grafickou pamětí. V prvním případě hrál prim čip Voodoo5, ve druhém ATI Radeon. Řešení použité v GeForce3 firmě NVIDIA opět zajišťuje dostatečný náskok.

Samozřejmě že nové funkce GeForce3 nebyly vyvinuty jen tak z ničeho. Výsledkem dlouhého společného vývoje s Microsoftem je nová verze DirectX 8.0. GeForce3 je s DirectX 8 plně kompatibilní a zároveň je to (prozatím) jediný produkt, který hardwarově implementuje Direct3D 8.0. Spolupráce firmy NVIDIA se společností discreet, vyvíjející 3d studio max, bude pro hráče i vývojáře počítačových her také velmi výhodná. Discreet pracuje na tom, aby všechny novinky čipu GeForce3 ovlivňující vzhled objektů mohl použít již uživatel 3d studia, a hlavně aby je rovnou ve 3d studiu viděl. Doposud uživatel 3d studia vůbec netušil, jak se jeho model zobrazí po přenesení do prostředí vyvíjené hry.

NOVINKY GEFORCE3

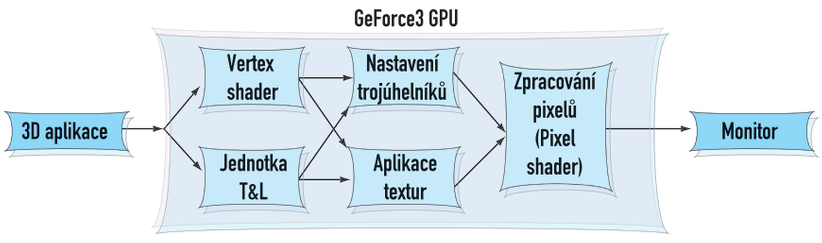
Podle materiálů firmy NVIDIA obsahuje GeForce3 čtyři „taháky" – programovatelný procesor pro zpracování vrcholů Vertex shader, programovatelný procesor pro zpracování pixelů Pixel shader, architekturu pro urychlení komunikace mezi čipem a pamětí Lightspeed Memory Architecture a hardwarově podporovaný celoobrazovkový antialiasing HRAA (High Resolution Antialiasing).

VERTEX SHADER

Vertex shader je pravděpodobně nejvýraznější novinkou GeForce3. Jde o plně programovatelný procesor pro zpracování vrcholů, umožňující aplikovat na každý vrchol program, který může jeho parametry téměř libovolně měnit.

Aby byl význam těchto slov zřejmější, pokusím se o stručné vysvětlení pojmu vrchol (angl. vertex) pro tvorbu 3D obrazu. Každý objekt ve 3D scéně je reprezentován množinou trojúhelníků. Aplikace (např. hra) pouze posílá do grafického čipu informace o vrcholech těchto trojúhelníků a všechny úkony nutné pro vytvoření obrazu již provádí grafický čip sám. Nejdříve dochází k transformaci vrcholů a aplikaci osvětlení na každý vrchol (to je úkol pro jednotku T&L nebo pro Vertex shader). Pak je pro každý vrchol trojúhelníku spočtena pomocí perspektivní projekce souřadnice pixelu, na nějž se daný vrchol zobrazí (nastavení trojúhelníků, triangle setup). Poté je pro každý pixel uvnitř trojúhelníku rozhodnuto, zda je viditelný na obrazovce (není zakrytý jiným objektem), a v případě, že je viditelný, je spočtena jeho barva aplikací textury, případně mlhy apod. Tyto úkony se provádějí pro každý trojúhelník ve scéně. Schéma tohoto procesu je na obr. 1.

Každému vrcholu je kromě jeho souřadnice přiřazeno mnoho dalších údajů (např. barva, souřadnice pro mapování textur, úroveň mlhy, lesklost atd.). Vertex shader může měnit libovolný z těchto údajů. K tomu má k dispozici sadu poměrně obecných, avšak velmi silných instrukcí. Každá instrukce má na vstupu jednu až tři čtveřice reálných čísel a výsledkem je jedna čtveřice reálných čísel. Program má k dispozici 16 registrů pro zápis a čtení a paměť konstant pouze pro čtení (typicky jsou v ní uloženy transformační matice). Každá instrukce se provádí jeden hodinový cyklus, program může obsahovat maximálně 128 instrukcí. V instrukční sadě není →



Obr. 1. Transformace scény na obraz uvnitř grafického čipu

Obr. 2. Pohyby delfína vznikají interpolací mezi několika tzv. klíčovými fázemi pohybu.

→ obsažena žádná instrukce pro cyklus, program se tedy provádí „lineárně“ – jedna instrukce po druhé, přesně v tom pořadí, jak byly zapsány v programu. To přináší obrovskou výhodu: jednotku Vertex shader lze snadno paralelizovat a je téměř jisté, že budoucí implementace DirectX 8.0 již budou obsahovat více paralelně pracujících jednotek Vertex shader (v GeForce3 je zatím pouze jedna).

Pro programátora je důležité vědět, že vstupem Vertex shaderu je jeden vrchol a výstupem opět jeden vrchol (transformovaný a osvětlený). Ve Vertex shaderu tedy žádné vrcholy nevznikají ani nezanikají.

Lepší než suchá teorie bude prohlédnout si pár příkladů.

SKINNING – KOST A KŮŽE

Při animaci postav se vychází z předpokladu, že postavička má kostru a ta je potažena „kůží“, která se přizpůsobuje pohybu kostry. V počítačové grafice to znamená, že postava je složena z určitého počtu kostí, které jsou vlastně reprezentovány transformační maticí (co kost, to jedna matice). Kůže je modelována jako plocha složená z trojúhelníků. K vrcholu každého trojúhelníku je přiřazena sada vah, které říkají, jak která kost ovlivňuje pohyb daného vrcholu. Při animaci postavičky je neustále nutné každý vrchol kůže násobit tolika transformačními maticemi, kolik kostí má vliv na pohyb onoho místa na těle umělé postavy. Tyto výpočty lze s GeForce3 provádět ve Vertex shaderu. NVIDIA uvádí, že lze použít až 32 matic (tedy 32 kostí). To je sice pravda, ale tím se vyčerpá celá kapacita 128 instrukcí Vertex shaderu. Realističtější tedy je předpokládat, že tento počet je nižší.

KEYING & MORPHING – JEDNODUŠŠÍ ANIMACE

Jednoduchý způsob, jak rozpohybovat objekt, je použít několik klíčových fází pohybu a interpolovat mezi nimi (interpolaci samozřejmě Vertex shader zvládne, má k tomu dokonce speciální instrukci) – obr. 2 ukazuje takto rozpohybovaného delfína.

PROCEDURÁLNÍ DEFORMACE

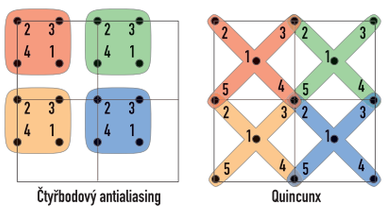
Představte si vlny na vodě po vhození kamenu – je to celkem jednoduchý tvar. A skutečně, pro-

placená inzerce

Obr. 3. Procedurální deformaci lze vypočítat pomocí programu ve Vertex shaderu.

Obr. 4. Pomocí Vertex shaderu lze aplikovat odraz i lom světla zároveň.

Obr. 5. Blinn bump mapping



Obr. 6. Quincunx je v porovnání se čtyřbodovým antialiasingem úspornější.

gram pro Vertex shader simulující takovou vlnu je celkem jednoduchý. Výsledek je na obr. 3.

ODRAZ A LOM SVĚTLA

Pomocí Vertex shaderu lze kombinovat lom i odraz světla zároveň. Dosud bylo možné použít buď jen odraz, nebo jen lom. Výsledek je úchvatný (obr. 4).

Tento výčet by mohl pokračovat ještě hodně dlouho, neboť možnosti Vertex shaderu jsou skutečně limitované více imaginací programátora než technickými omezeními. To NVIDIA ví, a proto se nespoléhá na představivost pouze svých inženýrů, ale obrátila se do světa. Společně se společností discreet vypsala soutěž o nejlepší efekt pro její program EffectBrowser. Výsledky jsou velice zajímavé a je možné si je prohlédnout na stránkách firmy NVIDA.

PŘÍKLAD PROGRAMU PRO VERTEX SHADER

Jako příklad programu pro Vertex shader jsem vybral program, který spočítá oboustranné osvětlení plochy – to znamená, že přední a zadní straně osvětlovaného trojúhelníku mohou být přiřazeny různé barvy. Ta správná z nich se vybere v závislosti na tom, zda uživatel vidí přední, nebo zadní stranu. Výsledkem je následující kód:

```
; skalární součin normály povrchu
; s vektorem spojujícím umístění pozorovatele
; s osvětlovaný vrcholem
dp3 r1, R_FACE_NORMAL, R_EYE_VECTOR

; pokud r1.x >= 0 (tj. přední
; strana je přilehlá k pozorovateli),
; do R6.x ulož 1, jinak ulož 0
sge r6.x, r1.x, c[C_ZERO].x
; pokud r1.x < 0 (tj. zadní strana
; je přilehlá k pozorovateli),
; do R6.y ulož 1, jinak ulož 0
slt r6.y, r1.x, c[C_ZERO].x

; spočítej intenzitu osvětlení pro přední stranu
dp3 r5.x, R_NORMAL, c[C_LIGHT_1_DIRECTION]
; spočítej intenzitu osvětlení pro zadní stranu
dp3 r5.y,-R_NORMAL, c[C_LIGHT_1_DIRECTION]

; barvu přední strany vynásob
; intenzitou osvětlní,výsledek ulož do r7
mul r7, r5.x, c[C_FRONTCOLOR]
; barvu zadní strany vynásob
; intenzitou osvětlní,výsledek ulož do r8
mul r8, r5.y, c[C_BACKCOLOR]

; vyber správnou barvu:
; vynuluj r7, pokud r6.x=0 (tj. zadní strana
; je přilehlá k pozorovateli)
mul r7, r7, r6.x
; vynuluj r8, pokud r6.y=0 (přední strana
; je přilehlá k pozorovateli), pak sečti r8 s r7
; a výsledek zapiš do výstupního registru
mad oD0, r8, r6.y, r7
```

PIXEL SHADER

Pixel shader je druhou jednotkou, kterou NVIDIA zařadila do přihrádky nfiniteFX engine. Jak je vidět na obr. 1, slouží ke zpracování jednotlivých pixelů, které se pak již zobrazují na obrazovku. Úlohou Pixel shaderu je zkombinovat informaci o barvě a osvětlení s texturami a vypočítat barvu každého pixelu. GeForce3 obsahuje čtyři jednotky Pixel shader, pracující plně paralelně.

Pixel shader je, stejně jako Vertex shader, programovatelný, avšak tato programovatelnost je v porovnání s Vertex shaderem značně omezená. Program pro Pixel shader je velice krátký (maximálně 12 instrukcí, 4 pro adresování textur a 8 pro míchání barev) a navíc instrukce jsou dosti jednoúčelově zaměřeny (na rozdíl od Vertex shaderu, jehož instrukce jsou obecné). Velkou výhodou Pixel shaderu je, že provádění operací může být řízeno výsledky Vertex shaderu.

Nechci vás však přesvědčovat o tom, že Pixel shader je špatný – seznam jeho možností je i tak dosti dlouhý. Zahrnuje stínové mapy (shadow maps), 3D textury do rozlišení $512 \times 512 \times 512$, aplikace až čtyř textur v jednom běhu, použití libovolného obrázku jako textury (např. Z-buffer se použije jako textura v případě stínových map) atd. Nebudu v tomto výčtu pokračovat, protože bez bližšího vysvětlení to nemá valnou cenu. Za všechny však alespoň jeden příklad.

BLINN BUMP MAPPING – HRBOLKY S ODRAZY

Podívejte se na obr. 5 a představte si, že by se každý hrbolek, který vidíte, měl modelovat pomocí trojúhelníků. Zdá se vám, že jich musí být tisíce? Naštěstí tomu tak není. To, co vidíte, jsou dva trojúhelníky (tvořící jeden čtyřúhelník), přičemž normála (tj. vektor kolmý k povrchu) je v každém bodu měněna tak, aby byla vytvořena iluze hrbolků. To zatím ale není nic nového – bump mapping zvládal i GeForce2. To, co je zde nového, jsou odrazy. Je jasné, že pro dokonalou iluzi hrbolků je třeba s normálou k ploše měnit i směr, z něhož se odráží světlo. A to Pixel shader hravě zvládá. Nejdříve od Vertex shaderu obdrží normálu k celé velké ploše. Tuto normálu pak v každém bodě moduluje tak, aby vznikla iluze hrbolů, a modulovanou normálu použije pro dotaz do textury, která reprezentuje odrážející se okolí.

LIGHTSPEED MEMORY ARCHITECTURE

Limitujícím faktorem dnešních grafických čipů je komunikace s pamětí, a to hlavně při vyšších rozlišeních a 32bitové barevné hloubce. Lightspeed Memory Architecture je soubor funkcí, jež by měly toto úzké hrdlo alespoň trochu rozšířit. Tři z nich, nazvané Loseless Z-Compression, Z-Oclussion Culling a Crossbar memory controller, byly popsány v článku Splašené pi- →

→ xely v Chipu 6/01. Další z nich se týká zpracování modelů s obecně tvarovaným povrchem.

PLOCHY VYŠŠÍHO STUPNĚ

I když komunikace nutná pro přenesení informací o vrcholech z aplikace do grafického čipu prozatím výraznější problémy nepřináší, určitě brzy bude, neboť objekty 3D scén jsou modelovány stále precizněji a k tomu je zapotřebí mnohem více trojúhelníků. Tento problém je řešen pomocí tzv. ploch vyššího stupně. Aby bylo dosaženo kvalitní aproximace hladké zaoblené plochy, je třeba použít velmi vysokého počtu trojúhelníků. Ale v počítačové grafice jsou již léta známy postupy, jimiž lze zaoblenou plochu snadno reprezentovat pomocí poměrně malého počtu tzv. kontrolních bodů (ano, mám na mysli Bézierovy plochy, splajny, příp. plochy NURBS).

Idea je jednoduchá – místo toho, aby byla oblá plocha nejdříve rozdělena na trojúhelníky a informace o vrcholech trojúhelníků se pak posílaly do grafického čipu, pošlou se do čipu pouze informace o kontrolních bodech (těch je mnohem méně než vrcholů) a rozdělení na trojúhelníky si provede čip sám. Zatím není jasné, jak kvalitní implementaci tohoto postupu GeForce3 obsahuje, ani zda je rozdělování na trojúhelníky dostatečně rychlé. Problémem může být také to, že existuje mnoho různých druhů ploch vyššího stupně, ale GeForce3 zpracovává jen některé.

HRAA

GeForce3 hardwarově implementuje tři režimy celoobrazovkového antialiasingu: 2× (každý obrazovkový pixel vzniká jako průměr dvou bodů), 4× (každý obrazovkový pixel vzniká jako průměr čtyř bodů) a nový paten-tovaný režim s prazvláštním pojmenováním Quincunx.

Podle materiálů firmy NVIDIA poskytuje Quincunx kvalitu režimu 4× pouze za „cenu“ režimu 2×. Pravda je, že kvalita obrazu je při použití Quincunxu skutečně téměř stejná jako u čtyřbodového antialiasingu a snímková frekvence se pohybuje asi v polovině mezi hodnotou získanou při použití dvoubodové a čtyřbodové metody.
Jak toho NVIDIA dosahuje, by měl lépe osvětlit obr. 6. Při čtyřbodovém antialiasingu se každý zobrazený pixel počítá jako průměr čtyř bodů, které jsou pro každý pixel různé. Proto je třeba spočítat barvu čtyřnásobného počtu bodů, než je počet pixelů. Avšak v případě Quincunxu se některé body „sdílí“ mezi různými pixely, a proto i když každý pixel vzniká průměrováním pěti bodů, je třeba spočítat pouze dvojnásobný počet bodů, než je počet pixelů. Ono sdílení bodů mezi pixely ale vede k tomu, že výsledný obraz je při použití metody Quincunx mírně rozmazaný (ale opravdu jen mírně).

ZÁVĚR

Jak jste asi z článku vycítili, nový čip od firmy NVIDIA se mi opravdu líbí. To, co se mi na něm líbí, je ohromné množství dobrých nápadů, které jsou do něho vloženy. Uvědomuji si však, že potenciálního kupce spíše zajímá, jaký výkonnostní růst za své investované peníze dostane. V současné době je situace pro GeForce3 dost nepříznivá. Výkonnostní nárůst oproti GeForce2 ULTRA je pro současné aplikace velmi malý, ale cena GeForce3 je stále velmi vysoká (v době psaní tohoto článku, v červenci, se nejlevnější karta s GeForce3 prodává za 14 000 Kč bez DPH). Správný čas pro zakoupení GeForce3 nastane v tu chvíli, kdy vám nějaký opravdový herní trhák při svém startu suše oznámí: „Vaše grafická karta neobsahuje Vertex shader, bude použita softwarová náhrada“ a vy zjistíte, že se tajemnými prostory šouráte rychlostí 5 snímků za sekundu.

Jaroslav Křivánek | jarda@slimak.cz

placená inzerce

V naší firmě... Belinea

BELINEA 101535
– 15" LCD displej
– maximální rozlišení: 1 024 x 768 @ 75 Hz
– H-kmitočet: 31 – 61 kHz
– TCO 99
Doporučená koncová cena (bez DPH): 15 990 Kč

BELINEA 101710
– 17" multimediální LCD
– velikost bodu: 0,264 mm
– maximální rozlišení: 1 280 x 1 024 @ 75 Hz
– H-kmitočet: 30 – 81 kHz
– TCO 99

BELINEA 103045
– 17" CRT
– velikost bodu: 0,26 mm
– maximální rozlišení: 1 600 x 1 200 @ 65 Hz
– H-kmitočet: 30 – 86 kHz
– TCO 99

BELINEA 106080
– 19" CRT NATURAL FLAT – DIAMONDTRON
– velikost bodu: 0,26 mm
– maximální rozlišení: 1 920 x 1 440 @ 75 Hz
– H-kmitočet: 30 – 110 kHz
– TCO 99

EXPRES SERVIS
Váš monitor Vám opravíme do 5 pracovních dnů.
Doprava do servisu a zpět zdarma.
www.belinea.cz

Penta
Strakonice: PENTA, tel.: 0342/369 111, e-mail: sales@penta.cz
Ostrava: PENTA, tel.: 069/6719 543, e-mail: sales@ostrava.penta.cz
www.penta.cz

KONSIGNA
Praha: KONSIGNA, tel.: 02/67 993 111, 02/67 993 200, e-mail: konsigna@login.cz; **Brno:** KONSIGNA, tel.: 05/51 47 165, e-mail: brno@konsigna.cz; **České Budějovice:** KONSIGNA, tel.: 038/733 0294, e-mail: c.budejovice@konsigna.cz; **Hradec Králové:** KONSIGNA, tel.: 049/553 7952, e-mail: hradec.kralove@konsigna.cz; **Olomouc:** KONSIGNA, tel.: 068/522 78 90, e-mail: olomouc@konsigna.cz; **Ostrava:** KONSIGNA, tel.: 069/623 7991, 069/623 79 82, e-mail: ostrava@konsigna.cz; **Plzeň:** KONSIGNA, tel.: 019/742 30 63, e-mail: plzen@konsigna.cz; **Ústí nad Labem:** KONSIGNA, tel.: 047/553 16 36, e-mail: usti@konsigna.cz
www.konsigna.cz