

Obsah

1	Architektura moderních procesorů	5
1.1	Hlavní součásti architektury počítačů založených na jednojádrových procesorech	6
1.2	Výpočty na CPU	10
1.2.1	Čísla ve formátu plovoucí řádové čárky	10
1.2.2	Výpočty s čísly v plovoucí řádové čárce na CPU	11
1.2.3	Nevýhody výpočtů na FPU	11
1.2.4	Vektorové zpracování – instrukční sady	11
1.3	Optimalizační techniky	12
1.3.1	Obecné optimalizace	13
1.3.2	Optimalizace zaměřené na cykly	14
1.4	Paralelní systémy a mechanismy	16
1.4.1	Flynnova taxonomie paralelních architektur	17
1.4.2	Metody programování paralelních systémů	18
1.4.3	Hodnocení kvality paralelních programů	21
1.4.4	Základní paralelní algoritmy	22
1.5	Architektura grafických procesorů	30
1.5.1	Rozdíly v architektuře CPU a GPU	33
1.5.2	Programování GPU a předchůdci CUDA technologie	35
2	Popis architektury NVIDIA CUDA	37
2.1	Programátorský model	38
2.2	Paměťový model	39
2.3	Spouštěcí model	40
2.4	Základy programování	43
2.4.1	Základní datové typy	44
2.4.2	Automaticky definované proměnné	45
2.4.3	Ošetření chyb	46
2.4.4	Funkce pro práci s globální pamětí	46
2.4.5	Sjednocený adresní prostor	49
2.4.6	Zjištění dostupných zařízení	50
2.4.7	Události	51
2.4.8	Komunikace v rámci warpu	52
2.4.9	Podporované matematické funkce	53
3	Programování v NVIDIA CUDA	57
3.1	Typy pamětí a jejich použití	57
3.1.1	Globální paměť	58
3.1.2	Sdílená paměť	61
3.1.3	Registry	62
3.1.4	Lokální paměť	63

3.1.5	Paměť pro konstanty	63
3.1.6	Paměť určená jen pro čtení	63
3.1.7	Specifikace umístění proměnných	64
3.1.8	Hlavní paměť hostitele	64
3.1.9	Paměť textur	66
3.1.10	<i>Surface Memory</i>	71
3.2	Synchronizace	71
3.2.1	Synchronizace v rámci bloku	72
3.2.2	Atomické operace	72
3.2.3	Synchronizace paměťových přístupů	73
3.3	Možnosti urychlení výpočtů	75
3.3.1	Využití více zařízení pro výpočty	75
3.3.2	Proudy	76
3.3.3	Vzájemná komunikace zařízení	80
3.3.4	Optimalizace fungování sjednoceného adresního prostoru	81
3.4	Dynamický paralelismus	82
3.4.1	Programátorský pohled	82
3.4.2	Omezení dynamického paralelismu	83
3.4.3	Příklady	84
3.5	Návrh programu pro GPU a optimalizace kódu	86
3.5.1	Využití obecných optimalizačních technik na GPU	87
3.5.2	Optimalizace CUDA kódu	88
3.5.3	Vytížení multiprocessorů	90
3.5.4	Maximální výkonnost SM	92
3.5.5	Sdružený přístup do globální paměti	92
3.5.6	Přístup do sdílené paměti	95
4	Ukázky řešení vybraných problémů	97
4.1	Výpočet histogramu	97
4.2	Paralelní redukce	99
4.2.1	Cena paralelní redukce	100
4.2.2	Vylepšení pomocí zmenšení využití sdílené paměti	102
4.3	Paralelní prefixový součet	102
4.3.1	Pracovně efektivní verze	102
4.3.2	Binární paralelní prefixový součet	105
4.4	Násobení matic	107
4.4.1	Verze se sdílenou pamětí	109
4.4.2	Optimalizace pomocí proudů	110
4.5	Kombinatorická úloha – problém batohu	111
5	Spolupráce s ostatními jazyky a nástroji	115
5.1	Překlad zdrojových kódů v CUDA (NVCC)	115
5.1.1	Nastavení CUDA kompilace	115
5.1.2	Práce se soubory	117
5.2	Spolupráce CUDA a OpenGL	118
5.3	Součásti CUDA SDK	119
5.3.1	cuBLAS	119
5.3.2	NVBLAS	119
5.3.3	cuFFT	119
5.3.4	Knihovna nvGRAPH	120
5.3.5	Knihovna cuRAND	120

5.3.6	Knihovna cuSPARSE	121
5.3.7	NPP	121
5.3.8	cuSOLVER	121
5.3.9	CUPTI	121
5.3.10	THRUST	121
5.3.11	Visual Profiler	123
6	OpenCL	125
6.1	Modely OpenCL	125
6.1.1	Model platformy	125
6.1.2	Prováděcí model	125
6.1.3	Programový model	126
6.1.4	Paměťový model	127
6.1.5	Práce s pamětí	128
6.1.6	Standardy OpenCL	128
6.2	Základy programování	129
6.2.1	Synchronizace	129
6.2.2	OpenCL 1.0 příklad	130
6.2.3	OpenCL 2.0 příklady	137
6.3	Rozdíly mezi OpenCL a CUDA	138
A	Vnitřní architektura grafických karet	141
A.1	Architektura karet firmy NVIDIA	141
A.1.1	Architektura multiprocessorů	141
A.1.2	Přehled GPU	142
A.1.3	Řada Geforce	142
A.1.4	Řada TEGRA /Jetson	143
A.1.5	Řada Quadro a NVS	143
A.1.6	Řada Tesla	143
A.1.7	Hardwarové parametry a limity	144
A.2	Vývoj architektury grafických procesorů v závislosti na CC	144
A.2.1	Architektura a vlastnosti CC 1.X karet	144
A.2.2	Architektura a vlastnosti CUDA CC 2.0 karet	145
A.2.3	Architektura a vlastnosti CC 2.1 karet	146
A.2.4	Architektura a vlastnosti CC 3.0 karet	146
A.2.5	Architektura a vlastnosti CC 3.5 karet	147
A.2.6	Architektura a vlastnosti CC 3.7 karet	147
A.2.7	Architektura a vlastnosti CC 5.0 karet	147
A.2.8	Architektura a vlastnosti CC 6.0 karet	148
A.2.9	Architektura a vlastnosti CC 6.1 a 6.2 karet	148
A.3	Architektura karet firmy ATI/AMD	148
A.3.1	Architektura VLIW4 a VLIW5	148
A.3.2	Architektura GCN	149
B	PTX ISA	151
B.1	Programový model	151
B.2	Instrukce pro celá čísla	151
B.3	Instrukce pro čísla v pohyblivé řádové čárce	152

C	OpenACC	153
C.1	Modely paralelismu	153
C.2	Základní konstrukce pro vyjádření paralelismu	154
C.3	Provádění kódu	154
C.4	Vliv kompilátoru	155
C.5	Práce s daty	155
C.6	Proměnné OpenACC prostředí	156
C.7	Další příkazy	156
C.8	Příklad - Jacobiho iterační metoda	157