*University of Toronto

Fizyka a obliczenia równoległe

*Paweł ARTYMOWICZ**

Komputery, największy wynalazek nowożytności, mają coraz więcej zastosowań. Jedne pracują w telefonach komórkowych i urządzeniach przenośnych. Innym – superkomputerom – zlecamy np. symulację historii wszechświata. Może kiedyś nauczymy je myśleć podobnie do tego jak sami myślimy. We wstępnym artykule z serii o współczesnych kierunkach w technikach obliczeniowych (zwłaszcza superkomputerowych) pokażemy, jak fizyka tranzystora umożliwiła technologii mikroprocesorowej podwoić prędkość komputerów więcej niż dwadzieścia kolejnych razy (tj. o czynnik $> 2^{20} \approx$ milion razy), umożliwiając szybki internet, smartfony i współczesną naukę obliczeniową, i dlaczego kontynuacja dotychczasowego wykładniczego rozwoju techniki komputerowej od trzynastu lat wymaga od programistów zasadniczo nowego podejścia: programowania współbieżnego procesorów wielordzeniowych.

Komputery są niezbędne do coraz szerszej listy zastosowań, od użytkowych (komunikatory osobiste, media i handel internetowy) i rozrywkowych (zaawansowane gry) do naukowych i inżynieryjnych, czyli od wymiany i analizy wartko rosnącego strumienia informacji, do symulacji rzeczywistości i prób prawdziwie inteligentnego przetwarzania informacji. Gdy wybiegniemy nieco w przyszłość, ale sądząc z rosnącego tempa rozwoju inteligencji maszynowej, w szczególności uczenia maszynowego – w całkiem niedaleką już przyszłość, nasze życie i światowy rynek pracy będą zmieniać się szybko dzięki tej jakościowo nowej fali komputeryzacji w stopniu nie mniejszym od tego, jak mechanizacja produkcji zmieniła społeczeństwa w epoce rewolucji przemysłowej. Niekoniecznie musi to oznaczać dominację maszyn i bezrobocie ludzi. Na przykład to, że pojazdy mechaniczne wyeliminowały konie i woźniców, nie oznaczało bynajmniej spadku zatrudnienia w transporcie. Pojawił się zawód kierowcy. Niedługo znajdzie się on na liście zawodów zagrożonych. Kierowców zastąpią komputery z wielkimi bazami danych, łącznością satelitarną, widzeniem i adaptacyjnym uczeniem maszynowym. W przewidywalnej przyszłości naukowcy czy lekarze i – niestety – również niektórzy politycy pójdą w ślady zapalaczy lamp gazowych (popularna w XIX wieku profesja).

Odpowiednikiem maszyny parowej, motoru rewolucji przemysłowej, jest dziś procesor komputera. Obliczenia równoległe jednego zadania na procesorach wielordzeniowych (tj. na wielu kalkulatorach naraz) mają zaś historyczny odpowiednik w metodzie produkcji masowej rozwiniętej w fabryce samochodów Forda w 1913 roku – zwiększają znacznie tempo obliczeń. Analogia jest o tyle niepełna, że każdy pracownik na taśmie produkcyjnej Forda mógł być (z założenia) mało wykształcony, w odróżnieniu od rzemieślnika tworzącego cały produkt samemu. Dzisiaj natomiast, jak zobaczymy, zachodzi konieczność przestawienia się programisty z myślenia i programowania sekwencyjnego na tworzenie programów współbieżnych albo równoległych (jak jednoczesna produkcja samochodów na równoległych taśmach produkcyjnych). A to wymaga nie mniejszych, lecz większych umiejętności niż programowanie zwykłego komputera von Neumanna, który pobiera i wykonuje jedną po drugiej instrukcje przy użyciu jednego kalkulatora (rdzenia obliczeniowego).

Postęp geometryczny

W latach 60. ubiegłego wieku współzałożyciel firmy Intel, Gordon Moore, sformułował empiryczną regułę geometrycznego postępu szybkości komputerów w ich kolejnych, regularnie co dwa lata (lub nawet nieco częściej) powstających pokoleniach procesorów komputerowych. Ściślej, prawo Moore'a mówi, że liczba tranzystorów na jednostkę powierzchni, tj. gęstość

Rozwiązanie zadania M 1530.
Niech $\ell(P)$ będzie długością łuku (mierzoną zgodnie z ruchem wskazówek zegara) łączącego P_0 z P, tzn. dla każdego $i = 0, 1, 2, \ldots, 2^n - 1$

$$\ell(P_i) = \frac{i(i+1)}{2} \pmod{2^n}.$$

Z założeń zadania wynika, że ℓ jest bijekcją zbioru wierzchołków 2^n-kąta i $\mathbb{Z} \cap [0, 2^n - 1]$.

Udowodnimy, że dla $n \geqslant 3$ zachodzi równość

$$\ell(\mathcal{E}) = \ell(\mathcal{F}) + 2^{n-2} \pmod{2^n},$$

czyli że zbiór $\mathcal{E}$ jest obrazem $\mathcal{F}$ przy obrocie o 90° wokół środka danego okręgu zgodnie z ruchem wskazówek zegara. Konkretnie, wykażemy, że dla każdego $P_i \in \mathcal{F}$

$$\ell(P_{f(i)}) \equiv \ell(P_i) + 2^{n-2} \pmod{2^n},$$

gdzie funkcja $f\colon \{i\colon P_i \in \mathcal{F}\} \to \{i\colon P_i \in \mathcal{E}\}$ określona jest następująco

$$f(i) = \begin{cases} 2^{n-1} + i & \text{dla } 2 \mid i, \\ 2^{n-1} - i - 1 & \text{dla } 2 \nmid i. \end{cases}$$

W obliczu definicji funkcji ℓ wystarczy więc sprawdzić, że dla każdego $i < 2^{n-1}$ liczba

$$g(i) := f(i)(f(i)+1) - i(i+1) - 2^{n-1}$$

jest podzielna przez 2^{n+1}. Rzeczywiście, bezpośrednio z definicji funkcji f otrzymujemy, że jeżeli $2 \mid i$, to

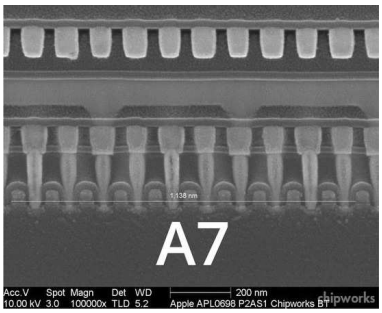
$$g(i) = 2^{2n-2} + i2^n \equiv 0 \pmod{2^{n+1}},$$

a jeżeli $2 \nmid i$, to

$$g(i) = 2^{2n-2} - (i+1)2^n \equiv 0 \pmod{2^{n+1}},$$

gdyż 2^{2n-2} dzieli się przez 2^{n+1} dla $n \geqslant 3$. Pozostaje bezpośrednio sprawdzić, że dla $n = 2$ rozważane zbiory także są przystające (odpowiednia izometria znów jest obrotem o 90°, ale w przeciwną stronę).

upakowania tranzystorów w płaskim układzie scalonym podwaja się co około dwa lata. Prawo jest szeroko znane, gdyż faktycznie tak zmieniała się ta liczba przez niesłychanie wiele pokoleń technologii procesorowych: ponad 20 (czyli przez ostatnie 40 lat) i ponieważ taki wykładniczy przyrost liczby bramek logicznych i komórek pamięci lokalnej w procesorze leży u podstaw błyskotliwej kariery komputera, a w miarę jak komputery stają się niezbędne, postępu technicznego cywilizacji. Spójrzmy na liczby opisujące typowy zaawansowany procesor główny (CPU, od ang. *Central Processor Unit*) w roku 1975 i 2015. Szybkość działania CPU mierzona jest liczbą najprostszych działań arytmetycznych zmiennoprzecinkowych na sekundę (jednostka zapisywana jako FLOP albo FLOP/s, to właśnie oznacza: floating point operation per second). Wynosiła w roku 1975 około 0,1 MFLOP (0,1 miliona FLOP). Czterdzieści lat później procesor CPU był już milion razy szybszy, 0,1–0,3 TFLOP (teraflop to 10^{12} działań/s), a koprocesory obliczeniowe osiągnęły 1 TFLOP w podwójnej precyzji obliczeń. Liczba tranzystorów w procesorach dochodzi do około 10 miliardów.



Rys. 1. Fragment procesora Apple A7, gdzie dziesięć tranzystorów ułożonych jest na odcinku 1138 nm. Procesor ten, znajdujący się w telefonach iPhone 5S, oparty jest na połączeniach o grubości 28 nm, tj. rzędu 70 odległości między atomami krzemu. (W latach 2010–2014 szerokość ścieżki procesorów Intel Corp. zmalała z 32 i 28 nm do 22 nm, a następnie do 14 nm). Szybko rosnący koszt produkcji tych niezawodnych, mikroskopijnych układów musi się producentom zwrócić. To zmusza ich do nieco rzadszej niż dawniej zmiany technologii produkcji. Pesymiści ogłosili wręcz, że pokolenie procesorów w technologii 10 nm będzie już ostatnim opłacalnym, czyli końcem technologii krzemowej. To niekoniecznie prawda. W odwodzie są też nowe substraty, np. grafen, prawdziwie 2-wymiarowa, heksagonalna sieć atomów węgla, która może kiedyś zastąpi krzem. Istnieje poważny, atomowo-kwantowy limit miniaturyzacji, związany z odległościami między atomami krzemu równymi 0,43 nm. Standardowy tranzystor nie będzie działał klasycznie po miniaturyzacji do kilku nanometrów i schemat funkcjonowania komputera musi być wtedy wymyślony od podstaw.

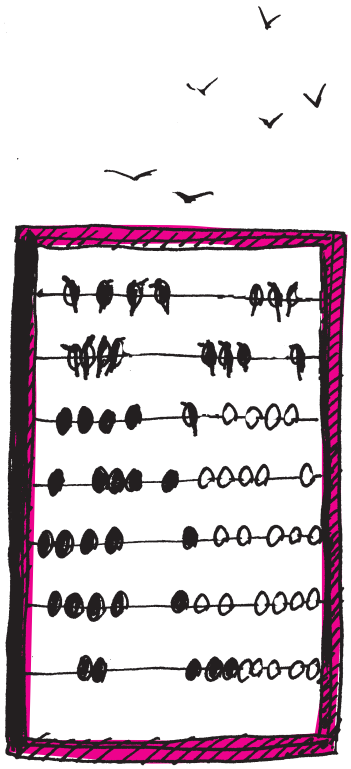
Prawo Moore'a nie było przypadkiem. Przeciwnie, było decyzją przemysłu elektronicznego, umożliwioną szczęśliwie przez prawa fizyki. Projektanci procesorów (np. Intel) miniaturyzowali podzespoły tak, by następna generacja procesora na tej samej powierzchni mieściła dwa razy więcej tranzystorów. To prawo Moore'a. Ale gdyby krok miniaturyzacji wymuszał np. dwa razy większe straty cieplne, to taka miniaturyzacja musiałaby być zatrzymana po kilku etapach miniaturyzacji. Kontynuowana, doprowadziłaby do tego, że obecnie komputer domowy zużywałby megawaty mocy, a telefon komórkowy rozładowywałby baterię w ułamku sekundy. Sprawdźmy zatem konsekwencje skalowania rozmiaru dla mocy pobieranej przez procesor, zamienianej na ciepło wskutek zjawiska oporności elektrycznej. Odprowadzenie znacznej ilości ciepła jest w istocie większym wyzwaniem, niż dostarczenie energii elektrycznej, choć koszty energii są niebagatelne. W przypadku superkomputerów są porównywalne z kosztem ich zakupu.

Bariera energii

Pierwsze mikroprocesory mało się grzały. Za to teraz zderzyliśmy się we wszystkich rodzajach obliczeń, od telefonii do centrum obliczeniowego, z barierą energetyczną. Pojedyncze procesory (CPU, koprocesory arytmetyczne i karty graficzne) nie mogą pobierać więcej niż $P = 200$ W, a najwyżej 300 W, inaczej byłoby trudno je zasilać w budynku mieszkalnym i równie trudno byłoby wentylować pokoje komputerowe. Na limit mocy napotykają też procesory w telefonach komórkowych i tabletach: długość pracy urządzenia bez ładowania baterii jest odwrotnie proporcjonalna do P, co nakłada na moc górne ograniczenie.

Energia elektryczna potrzebna do obliczeń szybko rośnie z częstotliwością f zegara procesora, jak i liczbą N tranzystorów. Bramka logiczna ma jeden lub więcej tranzystorów o efektywnej pojemności elektrycznej C (jest to stosunek ładunku do potencjału elektrycznego, $C = Q/V$). Ma także pewną oporność. Zmiana stanu bramki wymaga naładowania pojemności C poprzez opornik od potencjału zerowego do napięcia operacyjnego V. Przepływając przez źródło napięcia V, ładunek Q uzyskuje energię QV, czyli CV^2. Sumowanie przyczynków do elektrostatycznej energii potencjalnej tranzystora pokazuje, że niezależnie od wartości oporności bramki połowa energii źródła gromadzona jest na tranzystorze, a połowa zamieniana w czasie ładowania na ciepło. Tylko nieliczne procesory potrafią odzyskać tę pierwszą energię potencjalną, większość w końcu traci całość dostarczonej energii CV^2. Maksymalna moc rozpraszana w czasie obliczeń prowadzonych f razy na sekundę na N bramkach logicznych procesora zależy więc od f i V:

$$P \sim N\,C\,V^2\,f.$$

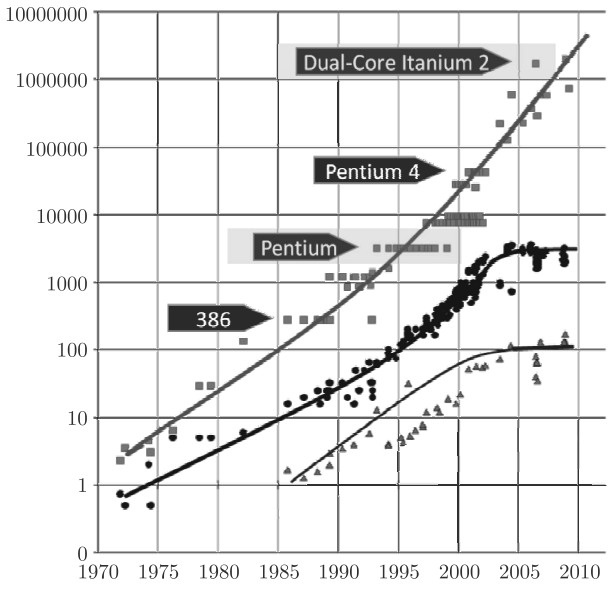


Jak zmieniają się parametry tranzystora przy zmianie generacji, tj. zmniejszeniu jego struktury o czynnik liniowy $\sqrt{2}$, a pola powierzchni o czynnik 2? Spójrzmy na kolejne czynniki powyższego wyrażenia na moc P. Liczba tranzystorów N podwaja się. Pojemność C to pewna stała materiałowa razy pole powierzchni elektrod (np. okładek kondensatora), podzielone przez odległość między nimi (to wynika z praw Coulomba i Gaussa w elektrostatyce). Ponieważ pole powierzchni spada dwukrotnie, a wszystkie odległości o czynnik skalowania $\sqrt{2}$, to pojemność C bramki spada o czynnik $\sqrt{2}$. Inżynierowie przy zmniejszeniu rozmiarów tranzystorów zmniejszali też o czynnik $\sqrt{2}$ różnicę potencjału V tak, by pole elektryczne (tzn. potencjał podzielony przez odległość elektrod) pozostało stałe. Przy niezmienionym V pole elektryczne za bardzo by w końcu wzrosło, niszcząc właściwości półprzewodnika. Zmiany stanu półprzewodników w tranzystorze zajmują pewien czas, który jest tym większy, im grubsze są warstwy półprzewodnika. I odwrotnie – zmniejszenie grubości pozwalało na zmniejszenie czasu przeładowania tranzystora. Dlatego częstotliwość f była zwiększana przy miniaturyzacji $\sqrt{2}$ razy. Zmiany wielkości po prawej stronie wyrażenia na P zebrane w jedną liczbę dają stałą wartość równą

$$(2)(1/\sqrt{2})(1/\sqrt{2})^2(\sqrt{2}) = 1.$$

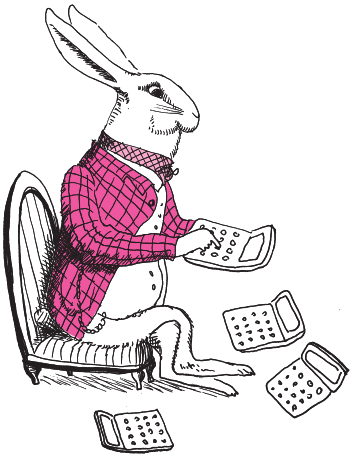
I tu właśnie leży tajemnica trzydziestoletniego sukcesu procedury skalowania: nie wymagało ono dostarczenia większej mocy tranzystorom na centymetrze kwadratowym powierzchni. Pobór mocy procesora rósł z innego powodu, takiego, że pole jego powierzchni w umiarkowanym tempie zwiększano. Liczba tranzystorów powiększała się o więcej niż czynnik 2. Można było użyć tak liczne tranzystory do zwiększania złożoności logiki procesora oraz na wewnętrzną, szybką pamięć zwaną podręczną (ang. *cache*). Było to potrzebne do maskowania narastającej od 2000 r. powolności pamięci operacyjnej RAM (*Random Access Memory*) w stosunku do CPU. Zastosowano coraz bardziej skomplikowane kolejki danych i instrukcji ściąganych przedwcześnie do pamięci podręcznej. Dzięki takim metodom szybkość zegara pomiędzy generacjami procesorów rosła nie o czynnik $\sqrt{2}$, lecz około dwukrotnie. Tempo rozwoju techniki obliczeniowej było fenomenalne.

(źródło – Intel)



Rys. 2. Wzrost w kolejnych latach liczby tranzystorów N w tysiącach (■), częstotliwości zegara procesora f w megahercach (●) i mocy elektrycznej jego zasilania P w watach (▲).

Idylla skalowania zakończyła się w latach 2003–2004, kiedy inne niż oporność elektryczna zjawisko dorównało dyssypacji energii w tranzystorze. Jest to prąd ucieczki (ang. *leakage current*), przeciekanie elektronów przez nominalnie nieprzewodzącą warstwę półprzewodnika. Zjawisko było drugorzędne przy dużych początkowo wartościach V, ale po kolejnym zmniejszeniu V dyssypacja energii związana z prądem ucieczki zaczęła dominować. Wynik był natychmiastowy, choć nienagłaśniany. Od tej pory utrzymywano $V \approx \text{const}$. Aby wydzielana moc cieplna P nie przekroczyła bariery energetycznej, konieczne też było praktyczne zamrożenie szybkiego uprzednio przyrostu częstotliwości f. Ilustruje to rysunek 2 pokazujący historię zmian N, f i P w latach 1970–2010. Po roku 2004 szybki przyrost f zakończył się. Dla oszczędności energii redukowano w niektórych urządzeniach f poniżej wartości 2–4 GHz osiąganych wcześniej. Tak jest np. w nowych procesorach arytmetycznych Intel Xeon Phi oraz procesorach graficznych (GPU), które mają obecnie $f \sim 1{-}1{,}5$ GHz. W cieniu bariery energetycznej wymuszającej $P < 300$ W, $f < 4$ GHz oraz $V \approx \text{const}$, prędkość obliczeń dość trudno jest dalej podnosić. Jest to absolutnie niemożliwe, jeśli zachowamy dawną architekturę procesora z tylko jednym, szybkim, centralnym kalkulatorem.



Myśleć równolegle

Świat (techniki obliczeniowej) uratowała po roku 2004 wielordzeniowość procesorów. To, że sprzedaje się teraz w sklepach procesory o coraz większej liczbie niezależnych rdzeni N_r, przyspiesza komputer proporcjonalnie do N_r, ale tylko pod warunkiem odpowiedniego dostosowania algorytmów. Gdyby nie działała nieubłagana fizyka, w tej chwili moglibyśmy liczyć na jednym rdzeniu CPU o $f = 30$ GHz, zamiast na 10 rdzeniach CPU o $f = 3$ GHz albo na 60 rdzeniach procesora Xeon Phi o zegarze 1,1 GHz. Programowanie byłoby tradycyjne, łatwiejsze, i wszystkie rodzaje zadań wykonywałyby się szybko. Ale w realnym świecie nie moglibyśmy tego robić w budynku mieszkalnym, gdzie nadmierny pobór prądu aktywowałby bezpieczniki!

Obecna ewolucja procesora polega na powielaniu w mniejszej skali przestrzennej rdzenia obliczeniowego o ograniczonej liczbie tranzystorów tak, aby obliczenia szły coraz większą liczbą równoległych torów (wątków programu). To pozwoliło procesorom kontynuować bicie rekordów sumarycznej mocy obliczeniowej (to temat następnego artykułu z tej serii). Ułatwiło procesorom, ale nie wszystkim programistom, i nie we wszystkich zastosowaniach. Rewolucja wielowątkowości została programistom narzucona przez inżynierów, ale – jak widzieliśmy – był bardzo istotny powód: energetyka tranzystora. Niektórzy opierają się do dziś idei programowania kart graficznych do celów obliczeniowych, mimo że do niektórych zastosowań w tej dekadzie będą to urządzenia zdecydowanie najszybsze (m.in. do uczenia maszynowego). Dla nich najlepszym rozwiązaniem mogą być procesory MIC (*Many Integrated Cores*), czyli Liczne Zintegrowane Rdzenie, jak Xeon Phi firmy Intel. Uruchamianie na nich programu, zwłaszcza wcześniej działającego programu sekwencyjnego, trwa krócej niż w przypadku GPU, urządzenia o podobnej liczbie N, ale odmiennej architekturze i hierarchii pamięci. W obu przypadkach nietrywialne programowanie nowoczesnych maszyn obliczeniowych jest jednak ciekawe, gdyż ich moce obliczeniowe nadal bardzo szybko rosną i pozwalają atakować dotychczas nierozwiązywalne problemy. Celem wytyczonym przez ministerstwo energii USA jest budowa do 2020 roku komputera robiącego 10^{18} działań arytmetycznych na sekundę (exaflop). Można też zbudować mini-superkomputer u siebie w domu i prowadzić obliczenia, np. dynamiki gazu, o czym opowie trzeci odcinek z tej serii.

A co dalej? Komputer kwantowy

Przewodniki w najnowszych CPU mają grubość 14 nm albo 40 warstw atomowych krzemu. Zmniejszenie ich 10 razy spowoduje, że procesor zacznie zachowywać się dziwnie, nieprzewidywalnie, gdyż elektrony i atomy będą przejawiały cechy kwantowe. Koncepcję obliczeń kwantowych sformułował fizyk, Richard Feynman, w 1981 roku. W wielu laboratoriach na świecie już teraz fizycy i inżynierowie dokonali w praktyce przeskoku do warstw jednoatomowych i koniecznego przy tym przejścia do fizyki kwantowej. Oczekujemy, że za około dziesięć lat zbudowane zostaną pierwsze użytkowe egzemplarze komputerów kwantowych. Cząstki reprezentować mogą jednocześnie wartości logiczne zera i jedynki na zasadzie superpozycji stanów kwantowych (w takim podwójnym stanie jest kot w słynnym eksperymencie myślowym Schrödingera). Jest więc pewne, że można będzie niesłychanie szybko sprawdzać wyniki wszystkich możliwych kombinacji zer i jedynek w obliczeniu probabilistycznym bądź kryptologicznym, jak i w niektórych problemach optymalizacji. Problem rozkładu dużej liczby naturalnej na czynniki pierwsze (ważny przy deszyfracji kodów) został kwantowo-algorytmicznie rozwiązany 23 lata temu. Nie jest tylko jasne, do ilu problemów spoza kombinatoryki i probabilistyki komputer kwantowy będzie się nadawał, czy będzie miał funkcjonalność komputera von Neumanna. Jeśli tak, to czeka nas zmiana paradygmatu obliczeń. Entuzjastyczne badania trwają, ale przez najbliższe 25 lat nie pozbywałbym się jeszcze komputera klasycznego.