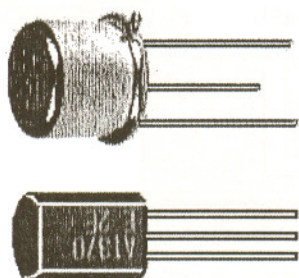


# Od tranzystora do rekonfigurowalnych komputerów

Ignacy Maciej

KUDEŁA



Rys. 1. Tranzystory.

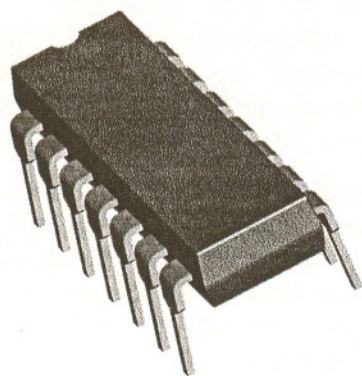
Sposób budowania cyfrowych urządzeń elektronicznych zmienił się bardzo w ciągu ostatniego ćwierćwiecza. Zmieniło się to, z czego się buduje te urządzenia, zmieniły się metody konstrukcji, ale także i to, co się konstruuje, jest inne.

Cyfrowymi urządzeniami elektronicznymi są zarówno komputery, jak też różnego rodzaju urządzenia automatyki przemysłowej, naukowej lub domowej. W latach 70. konstruktor takich urządzeń dysponował dosyć dużą gamą układów scalonych małej skali integracji (SSI). Zawierały one bramki logiczne i elementy pamięciowe, z których można było budować całe urządzenia. Zestaw bramek logicznych oraz elementów pamięciowych w miarę upływu czasu stawał się coraz bogatszy głównie po to, aby można było zaprojektować potrzebne urządzenie za pomocą mniejszej liczby elementów. Potem pojawiały się układy coraz większej integracji: średniej (MSI), dużej (LSI) i bardzo dużej (VLSI), pozwalając projektować urządzenia z jeszcze mniejszą liczbą podzespołów. Układy LSI i VLSI realizowały funkcje, które znajdowały szerokie zastosowanie, mogły być zatem produkowane w dużej ilości i w rezultacie ich cena mogła być niska. Typowe urządzenia cyfrowe budowano, używając kilku układów LSI lub VLSI, otaczając je grupą układów mniejszej integracji. Podstawową metodą projektowania urządzeń cyfrowych było rysowanie schematów elektronicznych. Układ SSI, MSI lub większej skali integracji był w takim schemacie reprezentowany przez blok (prostokątny lub inny), a jego funkcja opisana była – w przypadku skomplikowanych układów – na wielu stronach dokumentacji katalogowej. Zrozumienie działania urządzenia wymagało zatem znajomości schematu urządzenia (zaznajomienia się ze znaczeniem symboli graficznych w nim występujących), a także opisu katalogowego bardziej skomplikowanych układów.

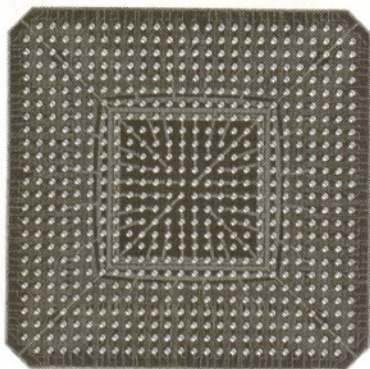
W latach 80. zaczęły pojawiać się programowalne układy logiczne. Zamiast składać schemat z kostek elementów logicznych (SSI lub MSI), można opisać funkcję, która ma być zrealizowana i spróbować zapisać ją do wnętrza takiego programowalnego układu. Te funkcje można było zmieniać, także wielokrotnie. Dzisiaj projektant dysponuje bardzo dużym zestawem takich układów. Istnieje wiele rodzin układów programowalnych różniących się wielkością, szybkością działania, sposobem programowania. Rodziny te są na ogół dosyć liczne, tak aby można było realizować za ich pomocą zarówno urządzenia proste (i tanie), jak i bardzo duże (i drogie). Duże układy programowalne mogą zawierać kilkaset doprowadzeń (nózek), setki tysięcy elementarnych elementów logicznych, które mogą być łączone w prawie dowolny sposób. Także sposób programowania (konfigurowania) takich układów może być bardzo wygodny – np. można je konfigurować przez komputer (przez sieć) na zmontowanym i działającym urządzeniu!

Wydaje się jednak, że najważniejsze (i najciekawsze) są zmiany w sposobie projektowania i sprawdzania funkcji, jakie te programowalne układy mają realizować. Współczesny projektant ma do dyspozycji różne zintegrowane środowiska programowe, za pomocą których może zaprojektować funkcje swojego urządzenia, załadować je do wybranego (lub dowolnego) programowalnego układu, a także zbadać, jak to, co zaprojektował, działa (w sensie algorytmu oraz szybkości). Jeżeli ta symulacja wygląda poprawnie, może oddać ten układ w ręce projektanta płytki drukowanej. Potem, po zmontowaniu całego urządzenia, może jeszcze zmodyfikować funkcje swojego układu (z wyjątkiem wyprowadzeń, które łączą ten układ z resztą urządzenia).

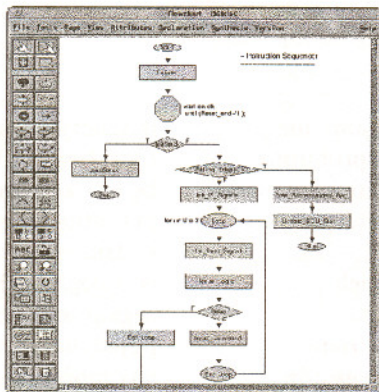
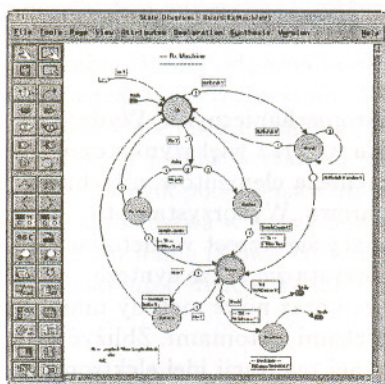
Są bardzo różne sposoby projektowania zawartości układów programowalnych: dla najbardziej konserwatywnych – można rysować schematy jak onegdaj, używając symboli graficznych odpowiadających funkcjom logicznym, ale można też opisać działanie za pomocą języka bardzo zbliżonego do języka C (z różnymi strukturami) lub też w końcu za pomocą grafu opisującego algorytm z elementami, które mogą być dowolnym sposobem opisu. Rezultat tej ewolucji



Rys. 2. Układ scalony SSI.



Rys. 3. Układ scalony VLSI.



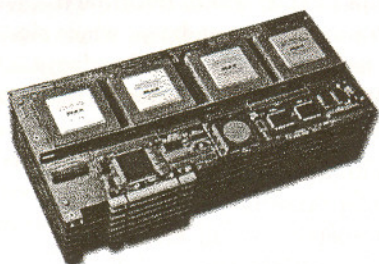
```

VHDL ENERGY - (SBEU)
File Tools Attributes Synthesis Version Help
Read Architecture Pop Top
architecture rev1 of COUNTn is
    signal count : std_logic_vector (width do
begin
    p1:
    process (clk)
    begin
        if (clk'event and clk = '1') then
            if (clr = '1') then
                count <= (others => '0');
            elsif (load = '1') then
                count <= d;
            else
                count <= count + "1";
            end if;
        end if;
    end process;
    y <= count;
end;
    
```

Rys. 4. Projektowanie układów programowalnych na przykładzie systemu Visual HDL™ firmy Summit Design Inc.

to łatwość opisu działania, znacznie zwiększona szybkość realizacji projektu, lepsza gwarancja poprawności. Znacznie zmniejsza się „odległość” między zlecającym projekt a jego realizatorem. Pozwala to na skuteczniejsze realizowanie projektu.

W rezultacie poszerza się zakres stosowania takich urządzeń. Jednym z najnowszych sposobów użycia programowalnych układów logicznych (CPLD i FPGA) są rekonfigurowalne komputery, których działanie jest określone nie przez program, ale przez konfigurację elementów logicznych. Jest to zestaw programowalnych układów logicznych połączonych między sobą często za pomocą także programowalnych elementów łączących. Ma on połączenie z komputerem, który pozwala na programowanie architektury i funkcji całego urządzenia. Tradycyjny komputer (wg koncepcji von Neumanna) pozwalał zrealizować dowolny algorytm; rekonfigurowalny komputer pozwala projektować komputer dla dowolnego algorytmu. Zapewnia to bardzo dużą szybkość działania.



Rys. 5. Rekonfigurowalny komputer XCITE-1000 firmy Axis Systems™.

Niezależnie od tych nowych zastosowań – współczesne urządzenie cyfrowe to kilka specjalizowanych VLSI uzupełnionych przez duży programowalny układ, który zapewnia wszelkie połączenia między VLSI. Dodatkowym elementem jest bezpieczeństwo praw autorskich twórców urządzenia – bez „treści” układu programowalnego urządzenie nie będzie działało!



## Zadania

Redaguje *Lukasz WIECHECKI*

**M 895.** Udowodnić, że suma kątów wszystkich ścian wielościanu wypukłego jest równa podwojonej sumie kątów wielokąta o tej samej liczbie wierzchołków.

Rozwiązanie na str. 14

**M 896.** Udowodnić, że każdy wielościan wypukły ma kąt trójsięenny lub ścianę trójkątną.

Rozwiązanie na str. 2

**M 897.** Udowodnić, że każdy wielościan wypukły ma ścianę o mniej niż sześciu bokach.

Rozwiązanie na str. 16

Redaguje *Ewa CZUCHRY*

**F 509.** Długość fali promienia czerwonego w wodzie jest równa długości fali promienia zielonego w powietrzu. Woda jest oświetlona światłem czerwonym. Jaki kolor widzi przebywający pod wodą nurek?

Rozwiązanie na str. 3

**F 510.** Gdy patrzymy na neonową reklamę świetlną, to litery czerwone zawsze wydają się być bliżej nas niż niebieskie czy zielone. Jak to wytłumaczyć?

Rozwiązanie na str. 3

**Uwaga:** Przypominamy Czytelnikom wzór Eulera, który prawdziwy jest dla wszystkich wielościanów wypukłych:

$$S - K + W = 2,$$

gdzie  $S$  oznacza liczbę ścian,  $K$  – liczbę krawędzi, a  $W$  – liczbę wierzchołków.