

SPRAWOZDANIE Z WYKONANIA ĆWICZENIA

IMIĘ, NAZWISKO: Igor Tryhub
Prowadzący pracownię: Rafał Szukiewicz
II rok studiów,
Informatyka Piątek 08:00 – 11:00

18 grudnia 2015 r.

Układ Arytmetyczno-Logiczny (ALU)

(34)

1. Opis teoretyczny

Bramka logiczna – element konstrukcyjny maszyn i mechanizmów (dziś zazwyczaj: układ scalony, choć podobne funkcje można zrealizować również za pomocą innych rozwiązań technicznych, np. hydrauliki czy pneumatyki), realizujący fizycznie pewną prostą funkcję logiczną, której argumenty (zmienne logiczne) oraz sama funkcja mogą przybierać jedną z dwóch wartości, np. 0 lub 1 (algebra Boole'a).

Podstawowymi elementami logicznymi, stosowanymi powszechnie w budowie układów logicznych, są elementy realizujące funkcje logiczne: sumy (alternatywy), iloczynu (koniunkcji) i negacji. Są to odpowiednio bramki OR, AND i NOT. Za pomocą dwóch takich bramek (OR i NOT lub AND i NOT) można zbudować układ realizujący dowolną funkcję logiczną, układy takie nazywa się układami zupełnymi.

Tabela wartości różnych rodzajów bramek:

p	q	FALŚ Z	p AND q	p NOT q	p XOR q	p OR q	p NOT R q	p XNO R q	NO T q	q → p	NO T p	p → q	p NAND D q	PRAWDA
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	0	1	0	1	0	1	0	1	0	1

Jednostka arytmetyczno-logiczna (z ang. *Arithmetic and Logical Unit* lub *Arithmetic Logic Unit*, **ALU**) to jedna z głównych części procesora, prowadząca proste operacje naliczanych całkowitych.

ALU jest układem cyfrowym, służącym do wykonywania operacji arytmetycznych (takich jak dodawanie, odejmowanie itp.), operacji logicznych (np. Ex-Or) pomiędzy dwiema liczbami oraz operacje jednoargumentowe takie jak przesunięcie bitów, negacja. ALU jest podstawowym blokiem centralnej jednostki obliczeniowej komputera.

Typowe ALU ma dwa wejścia odpowiadające parze argumentów i jedno wyjście na wynik. Operacje jakie prowadzi to:

- operacje logiczne AND, OR, NOT, XOR,
- dodawanie,
- przesunięcia bitowe o jeden bit, stałą liczbę bitów, czasem też o zmienną liczbę,

- często też, odejmowanie, negacja liczby, dodawanie z przeniesieniem, zwiększanie/zmniejszanie o 1
- dość często mnożenie i czasem dzielenie/modulo

Przyczyną dla której operacje te grupuje się w ALU jest to, że bramek logicznych potrzebnych do zaimplementowania wszystkich operacji z zestawu: dodawanie (z przeniesieniem i bez), odejmowanie (z przeniesieniem i bez), negacja liczby, zwiększanie i zmniejszanie o 1, AND, OR, NOT, XOR jest niewiele więcej od zaimplementowania samego dodawania. Szybkie mnożenie wymaga znacznie więcej, a dzielenie jeszcze więcej bramek w porównaniu do tych operacji.

Koder należy do klasy układów kombinacyjnych. Jest to układ posiadający n wejść oraz k wyjść ($n = 2^k$). Jego działanie polega na zamianie kodu "1 z n " na naturalny kod binarny o długości k . Służy do przedstawiania informacji tylko jednego aktywnego wejścia na postać binarną, ponieważ istnieje fizyczna możliwość aktywacji więcej niż jednego wejścia; musi istnieć możliwość uznania tylko jednego.

Dekoder należy do klasy układów kombinacyjnych. Jest to układ posiadający n wejść oraz $k=2^n$ wyjść. Jego działanie polega na zamianie naturalnego kodu binarnego (o długości n), lub każdego innego kodu, na kod "1 z k " (o długości k). Działa odwrotnie do koda, tzn. zamienia kod binarny na jego reprezentację w postaci tylko jednego wybranego wyjścia. W zależności od ilości wyjść nazywa się go dekodern 1zN.

Rejestr znaczników (flagowy) - rejestr przeznaczony do przechowywania dodatkowych cech wyniku operacji (np. znak, przekroczenie zakresu, znak parzystości, ...) w postaci tzw. flagi (znacznika). Każdy znacznik jest bitem w rejestrze, który wskazuje czy określony stan wystąpił. Znaczniki mogą być wykorzystywane przez procesor lub programistę na dwa sposoby:

- ustawienie znacznika dla zapamiętania określonego stanu po wykonaniu rozkazu
- testowanie znacznika celem umożliwienia decyzji o sposobie dalszego postępowania (przetwarzania danych)

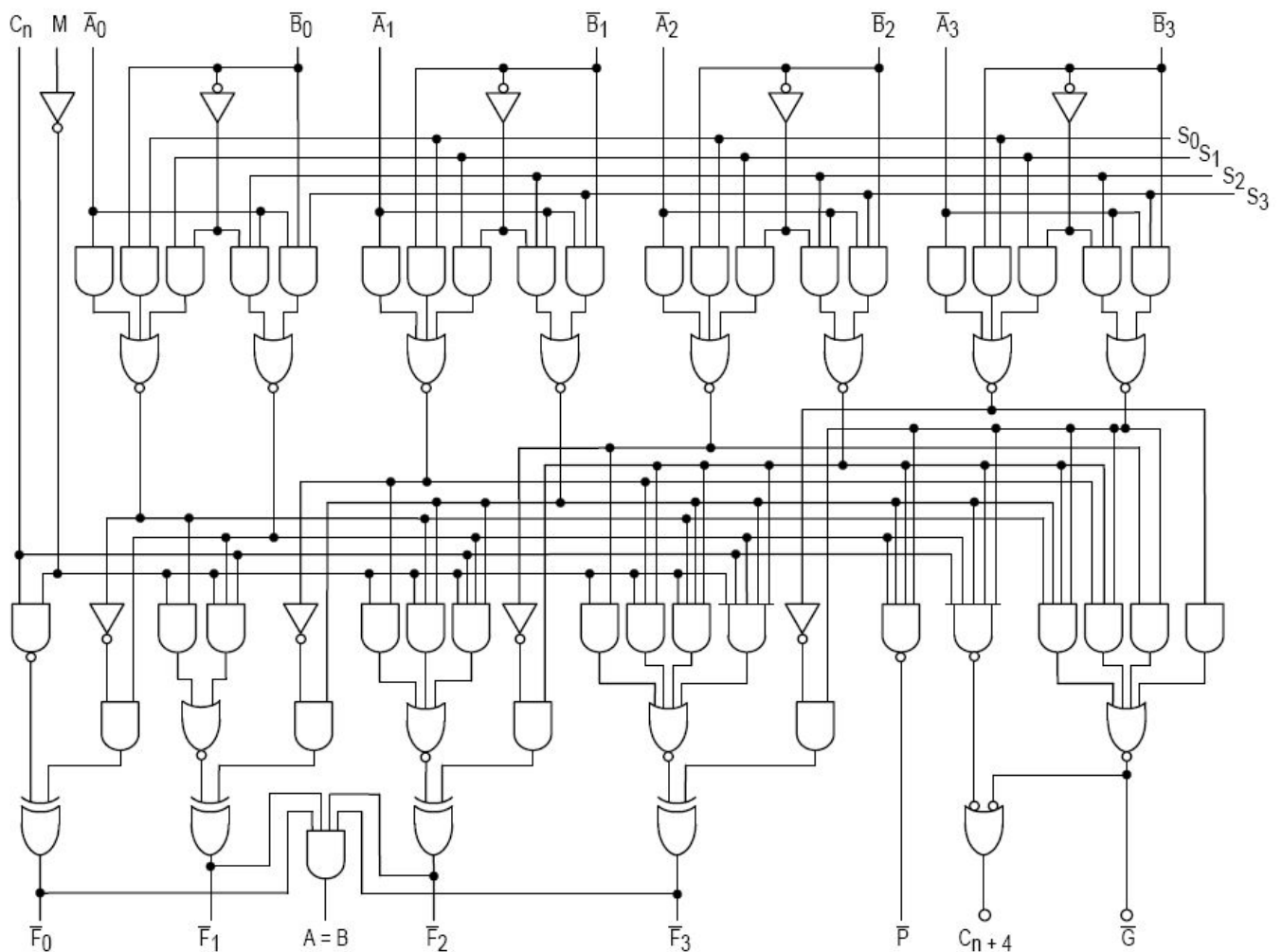
Znaczniki stanu:

- CF - flaga przeniesienia lub pożyczki, ustawiana przy przekroczeniu zakresu długości słowa, w którym zapisywany jest wynik (1 wystąpiło przeniesienie)
- PF - flaga parzystości (1) wskazuje na parzystą lub nieparzystą liczbę bitów o wartości jedynki
- ZF - flaga zera - sygnalizująca, że wynikiem ostatnio wykonywanej operacji jest zero
- PF - flaga przeniesienia pomocniczego jest wykorzystywana przy działaniach na liczbach w kodzie BCD. Ustawiana jest gdy następuje przeniesienie lub pożyczka z najstarszego bitu pierwszej tetrady wyniku
- SF - flaga znaku, ustawiana gdy najstarszy bit wyniku jest równy 1 (operacje liczb ze znakiem - kod U2)
- OF - flaga przepełnienia, sygnalizująca przekroczenie zakresu dla operacji arytmetycznych (operacje liczb ze znakiem - kod U2)

Znaczniki kontrolne:

- TF- znacznik pracy krokowej - ustawienie 1 pozwala na wykonanie przez procesor po wykonaniu każdego rozkazu przerwania i przejście do specjalnych procedur obsługi
- IF- znacznik zezwolenia na przerwanie - ustawienie 1 powoduje odblokowanie systemu przerwań w procesorze (0 - procesor ignoruje przerwania)
- DF - znacznik kierunku - umożliwia realizację przetwarzania łańcuchów (ciągu słów) przy rosnących adresach (1) lub malejących adresach (0)

74181 jest jednostką arytmetyczno-logiczną (ALU), realizowaną w 7400 serii TTL scalonego. Pierwszy kompletny ALU w jednym układzie był stosowany jako rdzeń arytmetyczny / logiczny procesorów wielu urządzeń.



Rys: Złożona logika obwody zintegrowanego obwodu 74181

Układ UCY 74181 ma czternaście wejść:

- Osiem wejść informacyjnych $A'_3 - A'_0$ i $B'_3 - B'_0$
- Cztery wejścia selekcyjne $S_3 - S_0$
- Wejście sterujące M
- Wejście przeniesienia/ pożyczki - C_0

Oraz osiem wyjść:

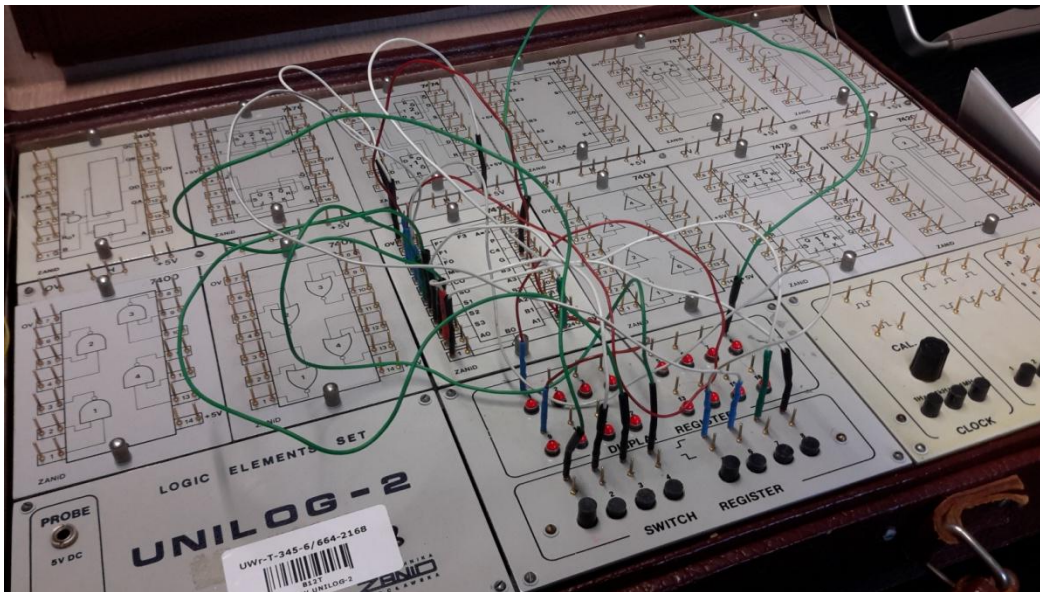
- Cztery wyjścia funkcyjne $F'_3 - F'_0$
- Wyjście komparatora $A=B$
- Wyjście przeniesienia - C_4
- Dwa wyjścia pomocnicze P' i G'

Układ ma wszystkie wejścia i wyjścia standardowe poza wyjściem komparatora które jest z otwartym kolektorem. Pojedynczy układ może realizować funkcje dla słów 4-ro bitowych jednak łącząc kolejne układy UCY 74181 można realizować funkcje na dłuższych słowach.

2. Wykonanie ćwiczenia

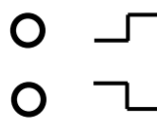
2.1. Odkręcając śrubki, przekonał się, że podwójne piny wyprowadzenia układu ALU 74181 są podłączone prawidłowo. Także okazało się, że po środku zestawu UNILog-2 przechodzi listwa zasilająca, która wyprowadza trzy sygnały:

- OV (inaczej - końcówka napięcia zasilania GND);
- H (stały poziom wyjściowego logicznego sygnału wysokiego);
- +5V (końcówka napięcia zasilania).



Rys.: Zestaw UNILog-2

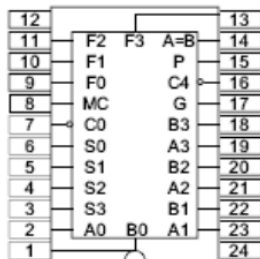
Także zbadaliśmy stany logiczne na zaciskach zestawu UNILog-2. Panel przełączników i wskaźników umieszczony jest w środkowej części układu. Składa się on z dwóch części: Switch Register oraz Display Register. W części Switch Register znajduje się 8 przełączników dwustabilnych, przeznaczonych do ręcznego ustawiania poziomów logicznych. Na górnym pinie wyjściowym występuje poziom L (low), jeżeli przełącznik jest zwolniony, oraz poziom H (high), jeżeli jest wciśnięty. Pin dolny daje negację sygnału wyjściowego przełącznika – zero logiczne przy wciśnięciu.



Rys.: Schemat przełączania pinów wyjściowych

W części Display Register znajdują się 16 wskaźników diodowych do monitorowania stanów logicznych w różnych punktach układów. Każda dioda emituje światło tylko gdy na jej odpowiadające wejście podany jest poziom logiczny H (high). Do budowy połączeń pomiędzy modułami i panelami technicznymi służą specjalne przewody połączeniowe o różnej długości wchodzące w skład zestawu.

2.2. Zapoznaliśmy się z przeznaczeniem różnych wyprowadzeń układu scalonego 74181, podłączyliśmy je do paneli przełączników i wskaźników.



Rysunek 1: Rozkład wyprowadzeń układu scalonego UCY 74181N.

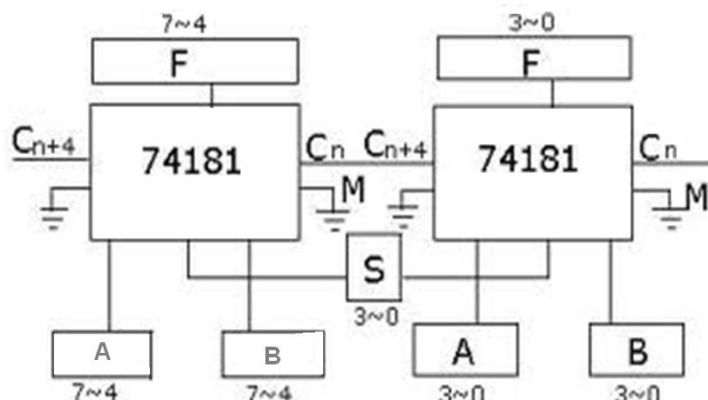
O ile poleceniem było sprawdzenie funkcji logicznych, operowaliśmy i sprawdzaliśmy poprawność jego działania tylko na najmłodszym bicie liczb A0 i B0. Więc podłączyliśmy wejścia A0, B0, S0, S1, S2, S3 C0, MC do wyjść przełączników stabilnych, a wyjścia F0, F1, F2, F3, A=B, C4 do wejść rejestru wskaźników. Zatem zrobiliśmy próbę sprawdzenia poprawność działania dla funkcji logicznych z tabeli działań tego układu.

Wejście selekcyjne				Funkcje logiczne	Funkcje arytmetyczne
S_3	S_2	S_1	S_0		
				$M_C = 1$	$M_C = 0, C_0 = 0$
L	L	L	L	$F = A$	$F = A$
L	L	L	H	$F = \overline{A \vee B}$	$F = A \vee B$
L	L	H	L	$F = \overline{A}B$	$F = A \vee \overline{B}$
L	L	H	H	$F = 0$	$F = -1$
L	H	L	L	$F = \overline{A}B$	$F = A + \overline{A}B$
L	H	L	H	$F = \overline{B}$	$F = (A \vee B) + \overline{A}B$
L	H	H	L	$F = A \oplus B$	$F = A - B - 1$
L	H	H	H	$F = A\overline{B}$	$F = \overline{A}B - 1$
H	L	L	L	$F = \overline{A} \vee B$	$F = A + AB$
H	L	L	H	$F = \overline{A} \oplus \overline{B}$	$F = A + B$
H	L	H	L	$F = B$	$F = (A \vee \overline{B}) + AB$
H	L	H	H	$F = AB$	$F = AB - 1$
H	H	L	L	$F = 1$	$F = A + A$
H	H	L	H	$F = A \vee \overline{B}$	$F = (A \vee B) + A$
H	H	H	L	$F = A \vee B$	$F = (A \vee \overline{B}) + A$
H	H	H	H	$F = A$	$F = A - 1$

Tabela 1: Tabela działania układu ALU.

Po pierwszym podłączeniu jednak nie udało się nam osiągnąć spodziewanych wartości znajdujących się w tabeli. Dlatego postanowiliśmy sprawdzić stabilność kontaktów na przewodach oraz sprawność diód. Wymieniliśmy przewody, które były podejrzenie luźne, oraz wykryliśmy, że jedna z diód faktycznie była zepsuta. Ale mimo to, nasz problem nie został rozwiązany. Dopiero po dokładniejszym przyjrzeniu się każdemu połączeniu, okazało się że wejścia selekcyjujące były podłączone w odwrotnej kolejności, niż te podane w tabeli (zamiast S3-S2-S1-S0 podłączyliśmy S0-S1-S2-S3). Po usunięciu tego błędu, udało się nam przekonać w poprawności działania funkcji logicznych.

2.3. Dla układów logicznych w ogóle, a także dla układu 74181 w szczególności, jest możliwe kaskadowe połączenie kilku modułów w celu uzyskania układu dla operacji na większych przedziałach danych wejściowych. Poniżej jest zaproponowany schemat połączeń dwóch układów 74181 dla otrzymania kaskadowego układu 8-bitowego.



Rys.: Schemat połączeń między dwoma układami 74181

Na tym schemacie widać, że 8-bitowe słowa na wejściu muszą być podzielone na dwie części po 4 bity. Wejścia S3, S2, S1, S0, MC muszą być takie same dla obu części, a więc muszą być zduplikowane na oba układy. Zastanawiania w takim kaskadowaniu potrzebowałoby zrealizowanie przeniesienia/pożyczki między dwoma układami dla operacji arytmetycznych. Nasza sugestia jest taka, żeby połączyć wyjście przeniesienia C4 z układu operującego na młodszych bitach z wejściem C0 układu operującym na starszych bitach danych wejściowych. W wyniku dostaniemy dane na wyjściach F3, F2, F1, F0 oraz F3', F2', F1', F0', które dla dalszego processingu należy ponownie złączyć w jedną całość. Szczęólnego traktowania wymagają też wyjścia A=B oraz A'=B'. Aby uzyskać poprawne dane dla 8-bitowych liczb, należy przepuścić te dwa wyjścia przez ALU AND, którego wyjście jest równe 1 tylko w przypadku gdy obie części liczb są takie same.

3. Wnioski

Na tej pracowni zbadaliśmy prosty 4-bitowy układ ALU 74181 przy pomocy treningowego zestawu UNILog-2. W czasie wykonywania tej pracowni nauczyłem się testować stany logiczne na zaciskach zestawu oraz analizowanie połączeń układów logicznych oraz wykrywać i usuwać możliwe błędy w ich działaniu. Przypomniłem sobie tabelę wartości różnych bramek logicznych. Utrwaliłem swoją wiedzę o niskich (low) i wysokich (high) poziomach logicznych napięcia układów cyfrowych. Oprócz tego, dowiedziałem się o zasadzie działań multipleksorów oraz rejestrach flagowych.