

Pracownia Konstrukcji i Eksploatacji Urządzeń Cyfrowych

Badanie układów cyfrowych

Badanie układów arytmetycznych

- **Badanie sumatora**
- **Badanie komparatora cyfrowego**

Badanie pamięci RAM

*opracowała:
mgr inż. Irena Hoja*

Sprawozdania wykonał:.....

Klasa:.....

Gdańsk 2016r.

1. Wprowadzenie

Cyfrowe układy scalone umożliwiające realizację podstawowych działań arytmetycznych, takich jak: dodawanie, odejmowanie, mnożenie i dzielenie – nazywa się **układami arytmetycznymi**.

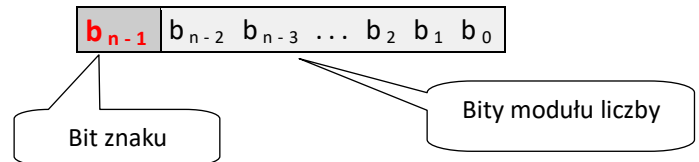
- Podstawowym układem arytmetycznym jest **sumator**, który dodaje dwie liczby dwójkowe. Za pośrednictwem operacji dodawania wielokrotnie powtórzonej jest dokonywane mnożenie, a przy zastosowaniu dodatkowych przekształceń również odejmowanie i dzielenie.
- Do układów arytmetycznych zaliczamy również: **komparatory** – porównujące liczby dwójkowe, **komplementery (transkodery** kodu ZM na zapis U1, U2, U9, U10), układy uniwersalne – **jednostki arytmetyczno – logiczne ALU**.

1.1. Metody zapisu liczb ze znakiem

Zapis w systemie znak-modułu - ZM

Liczba ZM składa się z dwóch części:

- bitu znaku
- bitów wartości liczby (modułu):



- Dla liczb dodatnich i zera **bit znaku** ma wartość **0**,
- dla liczb ujemnych i zera ma wartość **1**.
- „0” ma **podwójną reprezentację: 1000...000 lub 0000...000**

Zapis w systemie uzupełnień do 1 – U1

Format liczby U1 jest ustalony z góry – liczba zawiera określoną ilość bitów. W systemie tym wszystkie bity zapisu liczby posiadają swoje wagi (w ZM bit znaku nie posiadał wagi).

W zapisie U1 (uzupełnień do 1) MSB jest także bitem znaku : 0 – liczba dodatnia; 1 – liczba ujemna; ale w zależności od jego wartości dalsze bity mają różne znaczenie.

- Dla „0” (l. dodatnia) dalsze bity reprezentują liczbę w NKB.
- Dla „1” (l. ujemna) dalsze bity reprezentują moduł liczby ujemnej, w taki sposób, że zanegowane ich wartości odpowiadają modułowi tej liczby w NKB.

Wartości wag pozycji w zapisie U1							
waga	$-2^{n-1}+1$	2^{n-2}	2^{n-3}	...	2^2	2^1	2^0
cyfra	b_{n-1}	b_{n-2}	b_{n-3}	...	b_2	b_1	b_0

Bit znaku

Zapis w systemie uzupełnień do 2 – U2

Format liczby U2 jest ustalony z góry – liczba zawiera określoną ilość bitów. W zapisie U2 MSB jest także bitem znaku : 0 – liczba dodatnia; 1 – liczba ujemna; ale w zależności od jego wartości dalsze bity mają różne znaczenie.

- Dla „0” (l. dodatnia) dalsze bity reprezentują liczbę w NKB.
- Dla „1” (l. ujemna) dalsze bity reprezentują moduł liczby ujemnej. Moduł liczby ujemnej powstaje tak, że do zanegowanych pozycji słowa jest arytmetycznie dodawana jedynka i dopiero tak utworzone słowo odpowiada w NKB modułowi tej liczby.
- „0” ma **pojedynczą reprezentację: 0000...000**

Wartości wag pozycji w zapisie U2							
waga	-2^{n-1}	2^{n-2}	2^{n-3}	...	2^2	2^1	2^0
cyfra	b_{n-1}	b_{n-2}	b_{n-3}	...	b_2	b_1	b_0

Bit znaku

UWAGA! Przy realizowaniu operacji arytmetycznych musimy założyć określoną i jednakową dla wszystkich liczb długość ich reprezentacji binarnej. Jest to uwarunkowane technicznie (konkretny układ cyfrowy zawsze ma określoną długość przetwarzanych słów binarnych) i pociąga za sobą określone skutki, wymagające właściwego postępowania.

2. Cel ćwiczeń

Celem ćwiczeń jest:

- zapoznanie uczniów z właściwościami półsumatora, sumatora jednobitowego i czterobitowego oraz zastosowaniem sumatora do realizacji podstawowych operacji arytmetycznych,
- sprawdzenie działania komparatora cyfrowego '85,
- rozszerzanie wiadomości ucznia z zakresu kodowania i reprezentacji liczb w urządzeniach cyfrowych oraz w zakresie operacji arytmetycznych na liczbach dwójkowych.

3. Zakres materiału obowiązujący ucznia

- Znaczenie pojęć: półsumator, sumator, komparator.
- Kod dwójkowy naturalny.
- Kod BCD 8421.
- Metody zapisu liczb ze znakiem: zapis znak moduł (**ZM**), zapis uzupełnienie do 1 (**U1**), zapis uzupełnienie do 2 (**U2**).
- Podstawowe działania arytmetyczne w systemie dwójkowym.
- Wykonywanie działań logicznych oraz arytmetycznych na danych czterobitowych.

4. Ukształtowane umiejętności

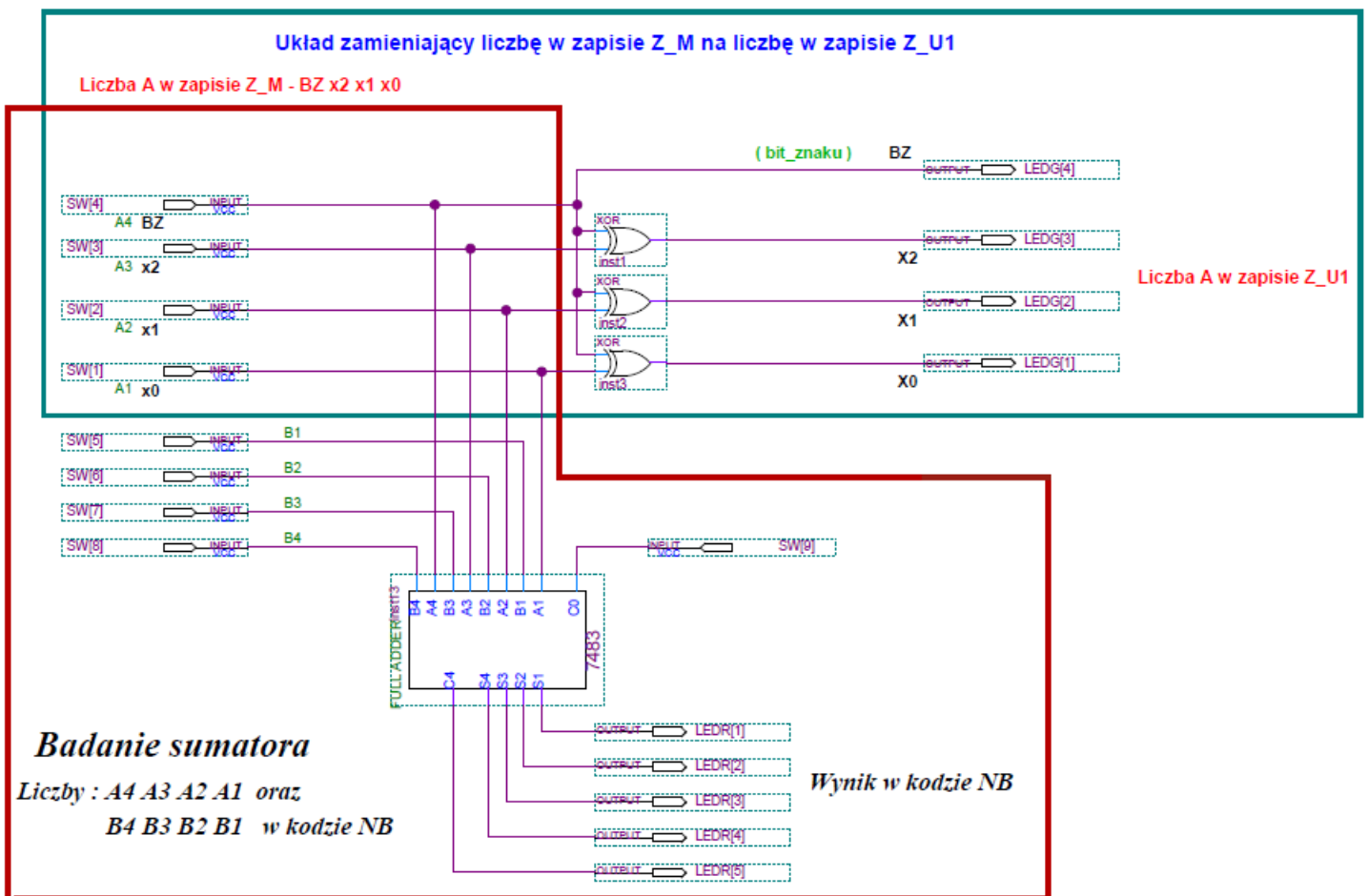
- Zapisywanie liczb w kodach U2, U1.
- Interpretowanie wartości liczby przedstawionej w kodzie U2 lub U1.
- Dokonywanie konwersji pomiędzy następującymi rodzajami kodów: dziesiętnym, dwójkowym naturalnym, U2, U1.
- Wykonywanie operacji arytmetycznych (dodawanie, odejmowanie, mnożenie) w ww. kodach.
- Wykorzystywanie sumatora scalonego '83 do budowania układów arytmetycznych.
- Wykorzystanie komparatora '85 w podstawowym układzie pracy.
- Rozróżnianie działań logicznych i arytmetycznych wykonywanych w układach cyfrowych.
- Wykonywanie operacji arytmetycznych i logicznych na liczbach dwójkowych.

5. Literatura - Głocki W.: Układy cyfrowe, WSiP – 1998

Imię i nazwisko:	Klasa:	Stanowisko:	Nr w dzienniku:
Skład grupy:.....			
Temat ćwiczenia: Badanie układów arytmetycznych: sumatora i komparatora			Data:
Przygotowanie do ćwiczenia	Wykonanie ćwiczenia	Sprawozdanie z ćwiczenia	
Pkt.:	Pkt.:	Pkt.:	
Suma punktów:			
Ocena z przeprowadzonego ćwiczenia:			Podpis nauczyciela:

ZADANIA POMIAROWE

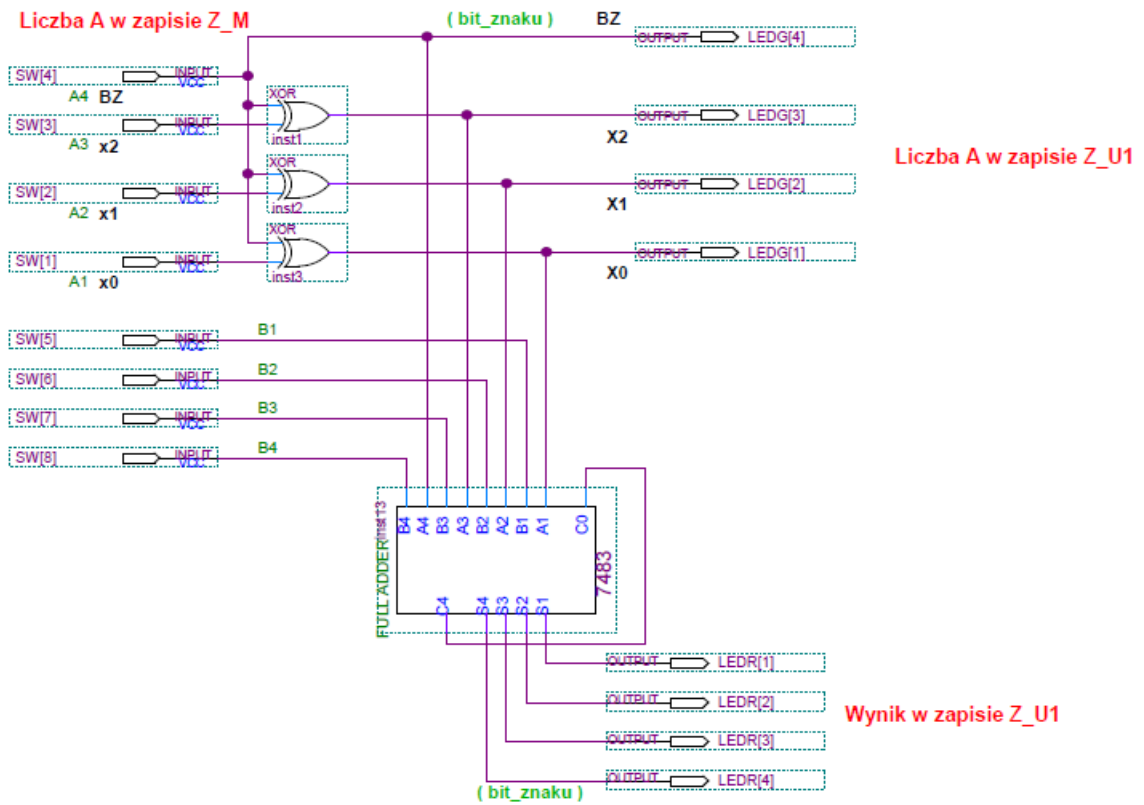
- Projekt nazwij **nr1_nr2_kl_SUM**
Narysuj w programie Quartus II (w edytorze graficznym) przedstawione poniżej schematy.
- Badanie sumatora i transkodera zapisu ZM (znak moduł) na zapis U1 (uzupełnienie do 1)



SUMATOR																
B				A				A+B								
DEC	B4	B3	B2	B1	DEC	A4	A3	A2	A1	C0	C4	S4	S3	S2	S1	DEC

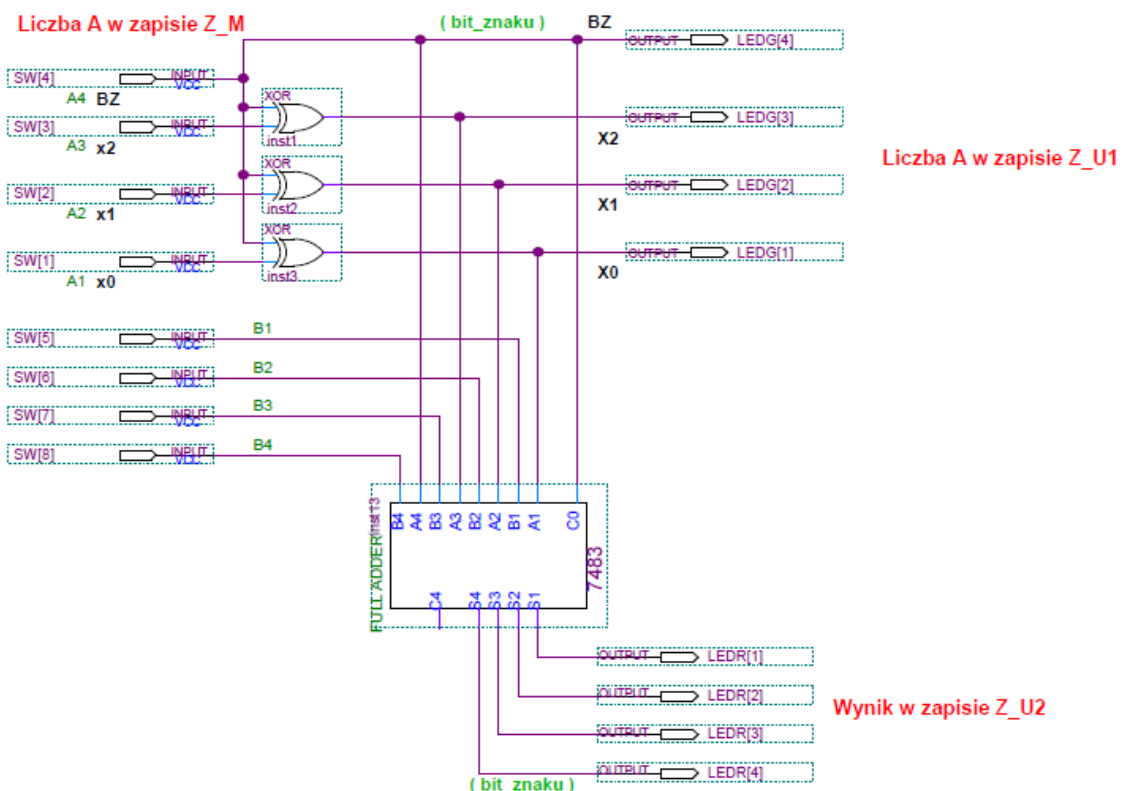
Transkoder zapisu ZM na zapis U1							
DEC	BZ	x2	x1	x0	X2	X1	X0
+	0						
+	0						
-	1						
-	1						
+	0						
+	0						

3. Badanie układu dodającego – odejmującego liczby zapisane w kodzie Z_U1



B				A				A+B							
DEC	B4	B3	B2	B1	DEC	A4	A3	A2	A1	S4	S3	S2	S1	DEC	
+	0				+										U1
+	0				+										ZM
+	0				-										U1
+	0				-										ZM
+	0				-										U1
+	0				-										ZM

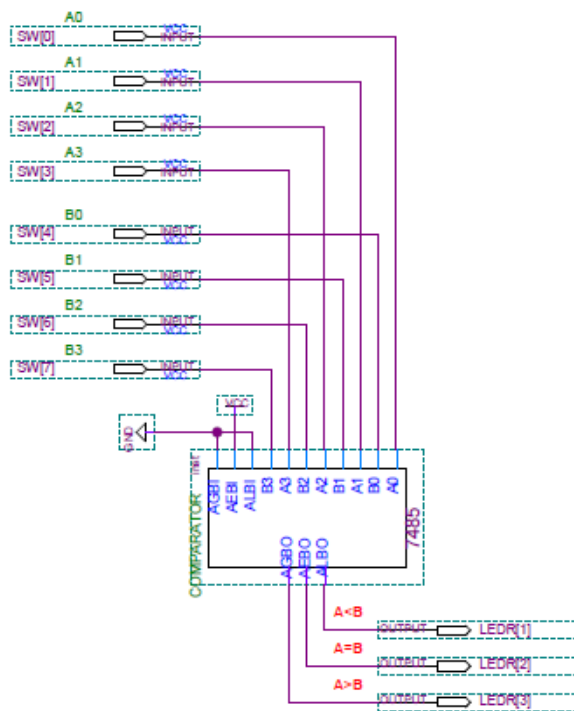
4. Badanie układu dodającego – odejmującego liczby zapisane w kodzie Z_U2



B					A					A+B					
DEC	B4	B3	B2	B1	DEC	A4	A3	A2	A1	S4	S3	S2	S1	DEC	
+	0				+										U2
+	0				+										ZM
+	0				-										U2
+	0				-										ZM
+	0				-										U2
+	0				-										ZM

5. Badanie komparatora

BADANIE KOMPARATORA



B					A					wyjścia		
DEC	B3	B2	B1	B0	DEC	A3	A2	A1	A0	A > B	A = B	A < B

6. Wnioski:

Badanie pamięci półprzewodnikowych RAM

• Wprowadzenie

Pamięci są układami służącymi do przechowywania informacji. Informacja jest pamiętana w postaci słów binarnych.

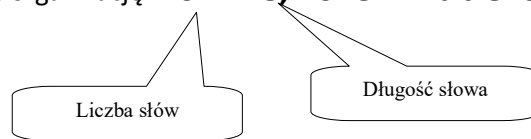
Pamięci półprzewodnikowe dzieli się na:

- pamięci o dostępie swobodnym RAM (Random Access Memory)
- pamięci stałe ROM (Read Only Memory).

Podstawowym parametrem pamięci jest jej **pojemność**, określająca, jak wiele informacji można w niej przechować. **Pojemność** określa się w bitach [b], kilobitach [Kb], megabitach [Mb] {1Kb = 2^{10} b, 1Mb = 2^{20} b, 1Gb = 2^{30} b} lub bajtach – 1B = 8 b {1KB = $2^{10} \cdot 8$ b = 8192 b}.

Organizacja pamięci mówi nam o sposobie dostępu do informacji czyli o liczbie wejść adresowych i długości pamiętanego słowa (liczbie bitów dostępnych pod jednym adresem).

Np.: pamięć o pojemności 8Kb może mieć organizację: **1024 · 8, 2048 · 4 lub 8192 · 1**



Pamięci RAM ze względu na budowę dzielą się na:

- **SRAM** - (Static RAM), pamięć statyczna. Cechuje ją bardzo krótki czas dostępu do poszczególnych komórek nieulotność w czasie pracy.
- **DRAM** - (Dynamic RAM) pamięć dynamiczna. Pamięć ta jest wolniejsza niż pamięć SRAM, a w dodatku jest ona ulotna. Aby pamięć ta nie utraciła danych trzeba ją odświeżać z częstotliwością co najmniej kilkaset Hz.
- **SDRAM** - (Synchronous Dynamic RAM) pamięć dynamiczna, synchroniczna. Pamięć ta jest podobna do pamięci DRAM, z tym że dostęp do komórek pamięci jest zsynchronizowany z zewnętrznym zegarem taktującym np. procesor.

Podział ze względu na **dostęp**:

- **FPM RAM** - (Fast Page Mode RAM), pamięć ta zorganizowana jest w strony, przy czym najszybciej realizowany jest dostęp do kolejnych komórek w obrębie strony.
- **EDO RAM** - (Extended Data Output RAM), jest to pamięć w przypadku której w czasie odczytu danej komórki, może zostać pobrany adres następnej.
- **BEDO RAM** - (Burst EDO RAM), w przypadku tej pamięci zamiast jednego adresu pobierane są cztery, przy czym na magistralę wystawiany jest tylko pierwszy co znacznie zwiększa szybkość dostępu.

Pamięć typu RAM powinna posiadać następujące **wejścia**:

- Wejścia adresowe
- Wejścia danych
- Wejścia wyboru operacji W/\bar{R} (zapis / odczyt)
- Wyjścia danych
- Selekcji układu CS
- Zegarowe (w pamięciach synchronicznych)

Imię i nazwisko:	Klasa:	Stanowisko:	Nr w dzienniku:
Skład grupy:.....			
Temat ćwiczenia: Badanie pamięci półprzewodnikowych RAM			Data:
Przygotowanie do ćwiczenia	Wykonanie ćwiczenia	Sprawozdanie z ćwiczenia	
Pkt.:	Pkt.:	Pkt.:	
Suma punktów:			
Ocena z przeprowadzonego ćwiczenia:			Podpis nauczyciela:

1. Cel ćwiczenia

Celem ćwiczenia jest zapoznanie ucznia ze sposobem adresowania pamięci, zapisem i odczytem informacji, jak również ze zwiększaniem pojemności pamięci.

2. Wstęp teoretyczny

- Co to jest pamięć RAM i ROM? Co oznacza pojęcie : pamięć o dostępie bezpośrednim? Czy pamięci RAM, ROM należą do klasy pamięci o dostępie bezpośrednim?

.....

- Zdefiniuj takie pojęcia jak pojemność pamięci i organizacja pamięci

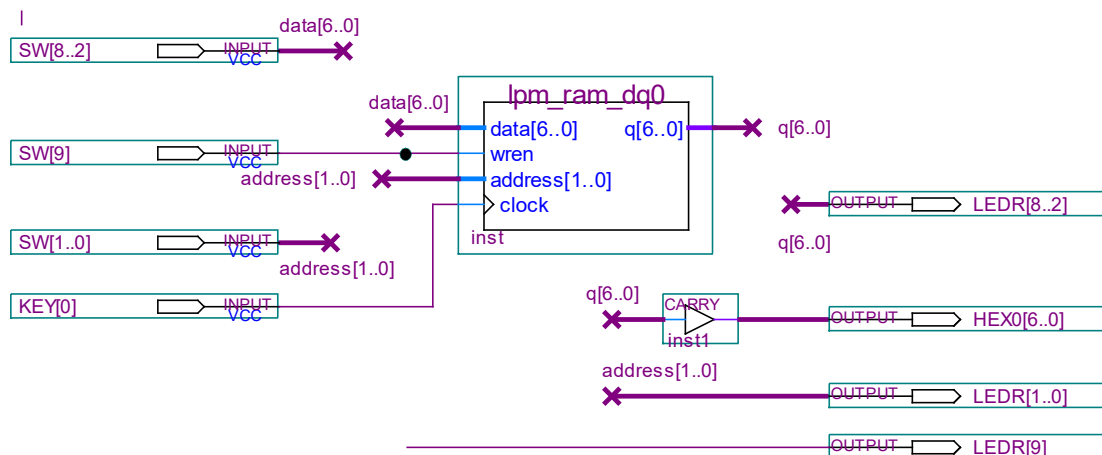
.....

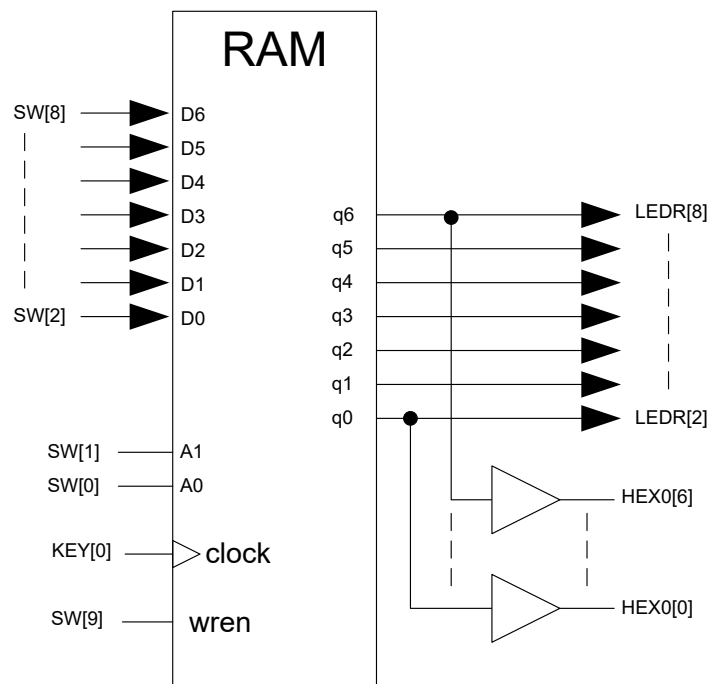
3. Zadania pomiarowe

3.1. Zaimplementuj w układzie FPGA pamięć RAM.

Projekt nazwij **nr1_nr2_kl_RAM**

Narysuj w programie Quartus II (w edytorze graficznym) przedstawiony poniżej schemat.





wren = 1 zapis
wren = 0 odczyt

3.2. Określ organizację i pojemność pamięci przedstawionej na schemacie.

Mając do dyspozycji 9 przełączników SW [8..0] umożliwiających wprowadzenie danych do układu, pamięci o jakiej organizacji i pojemności mógłbyś zbudować na układzie **lpm_ram_dq()**

Liczba wej. adresowych	Długość pamiętanego słowa [b]	Organizacja	Pojemność [b]
2	7	4 * 7	28
3	6	8 * 6	48
4	5	16 * 5	80
5	4	32 * 4	128
6	3	64 * 3	192
7	2	128 * 2	256
8	1	256 * 1	256

3.3. Badanie pamięci

3.3.1. Zapis pamięci (wprowadzanie danych):

- SW[9]=wren=1 – zezwolenie na zapis
- Ustawić adres na SW[1..0]
- Ustawić dane na SW[8..2]
- Podać impuls taktujący

3.3.2. Odczyt pamięci:

- SW[9]=wren=0 – zezwolenie na odczyt
- Ustawić adres na SW[1..0]
- Podać impuls taktujący
- odczytać dane na LEDR[8..2] i na wyświetlaczu HEX0

3.3.3. Wyniki badań zapisać w tabeli.

3.3.4. Zapisać w pamięci kolejne znaki

Lp.	SW[9]=wren=1 [zapis]									SW[9]=wren=0 [odczyt]							
	adres		dane [d]							wyjścia [q]							
			g	f	e	d	c	b	a								
	SW [1]	SW [0]	SW [8]	SW [7]	SW [6]	SW [5]	SW [4]	SW [3]	SW [2]	LEDR [8]	LEDR [7]	LEDR [6]	LEDR [5]	LEDR [4]	LEDR [3]	LEDR [2]	HEX0
0	0	0	1	0	0	0	0	0	0								0
1	0	1	1	1	1	1	0	0	1								1
2	1	0	0	1	0	0	1	0	0								2
3	1	1	0	1	1	0	0	0	0								3
																	4
																	5
																	6
																	7
																	8
																	9

3.4. Wnioski: