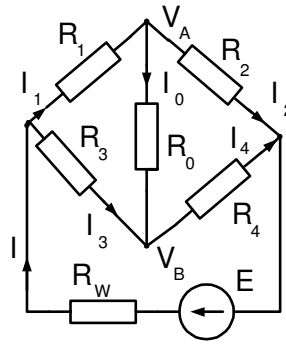


Etap A - Test

1. W poniższym układzie mostka prawdziwe są zależności:

- a. $I_1 + I_0 + I_3 = 0$
- b. $-I_1 - I_3 + I_2 + I_4 = 0$
- c. $I_1 I_4 = I_3 I_2$
- d. $I_0 R_0 = \frac{E}{R_w} (R_1 + R_2 + R_3 + R_4)$



2. Po zamianie miejscami rezystancji ($R_1 \leftrightarrow R_3$ i $R_4 \leftrightarrow R_2$) pomiędzy nowymi prądami (primowanymi) a „starymi” prądami, zachodzą zależności:

	a	b	c	d
$I_0' =$	Trzeba na nowo przeliczyć	I_0	$- I_0$	$- I_0$
$I_1' =$		I_2	I_4	I_3
$I_2' =$		I_1	I_3	I_4
$I_3' =$		I_4	I_2	I_1
$I_4' =$		I_3	I_1	I_2

3. $| I_0 | = 0$ gdy pomiędzy mocami w gałęziach powyższego mostka zachodzi zależność:

- a). $P_2 - P_1 = P_3 - P_1$
- b). $P_1 P_2 = P_3 P_4$
- c). $P_1 P_3 = P_2 P_4$
- d. $P_1 P_4 = P_2 P_3$

4. Schemat jak w zadaniu 1. Wiemy, że: $P_1 = 8 \text{ W}$, $P_2 = 45 \text{ W}$, $V_A = 15 \text{ V}$, $R_1 = 2 \Omega$. Prąd I_0 wynosi:

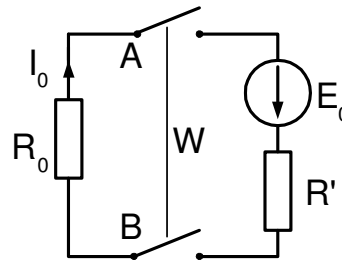
- a. 1 A
- b. 0 A
- c. - 1 A
- d. za mało danych

5. W układzie jak w zadaniu 1, wartości elementów wynoszą:

$R_1 = 7 \Omega$, $R_2 = 1 \Omega$, $R_3 = 4 \Omega$, $R_4 = 6 \Omega$, $R_0 = 1 \Omega$, $R_w = 0 \Omega$, $E = 18 \text{ V}$.

Podaj parametry E_0 , R' dwójnika widzianego z zacisków A – B. Po załączeniu przełącznika W popłynie prąd I_0 .

	a	b	c	d
E_0	$\frac{153}{16}$	6	$\frac{171}{20}$	7
R'	$\frac{121}{16}$	2	$\frac{131}{40}$	6

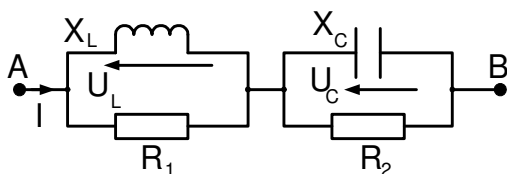


6. Ile czasu zajmie naładowanie kondensatora 2,2 μF do napięcia 10 V idealnym źródłem prądu o wartości 50 mA:

- a. 440 μs
- b. 1,1 ms
- c. 2,2 ms
- d. 4,4 ms

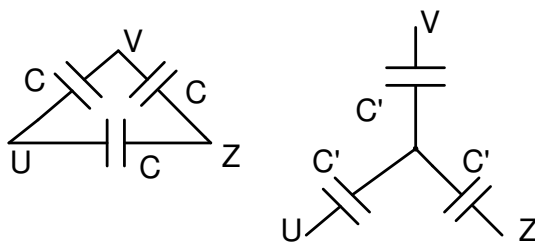
7. Dwójnik AB jest zasilany napięciem sinusoidalnie zmiennym (stan ustalony). Kąt przesunięcia fazowego pomiędzy napięciem a prądem wynosi ($|X_L| = R_1$, $|X_C| = R_L$, $2R_1 = R_2$):

- a. 0
- b. $\frac{\pi}{4}$
- c. $|\text{arc tg } 0,5|$
- d. $\frac{\pi}{4} - |\text{arc tg } 0,5|$



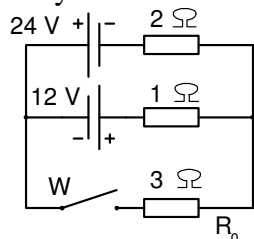
8. Po zamianie trójkąta z pojemnościami C na równoważną gwiazdę, wartości pojemności C' wynoszą:

- a. **3C**
- b. $\frac{2}{3} C$
- c. $2C$
- d. $\frac{1}{3} C$



9. Dwa źródła napięcia połączono jak na rysunku. Po zamknięciu wyłącznika W przez R_0 popłynie prąd:

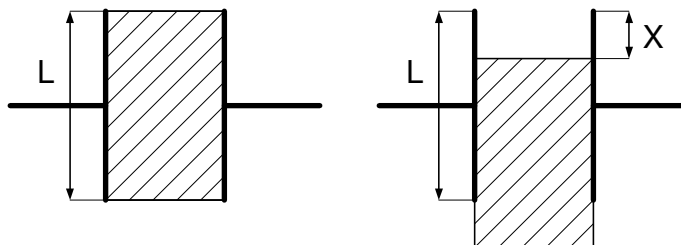
- a. 12 A
- b. 8 A
- c. 4 A
- d. **0**



10. W kondensatorze płaskim wysunięto część dielektryka o przenikalności względnej ϵ_r . Określ w procentach, jaką część dielektryka należy wysunąć, aby pojemność zmniejszyła się n – krotnie [$n \in (1, \epsilon_r)$]. Wartość

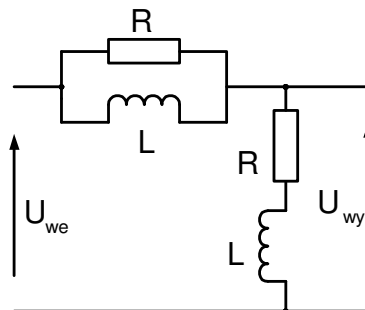
$\frac{X}{L} \cdot 100\%$ wynosi:

- a. $\frac{(\epsilon_r - 1)(n - 1)}{n\epsilon_r} \cdot 100\%$
- b. **$\frac{\epsilon_r(n - 1)}{n(\epsilon_r - 1)} \cdot 100\%$**
- c. $\frac{\epsilon_r n}{(n - 1)(\epsilon_r - 1)} \cdot 100\%$
- d. $\frac{(\epsilon_r - 1)n}{(n - 1)\epsilon_r} \cdot 100\%$



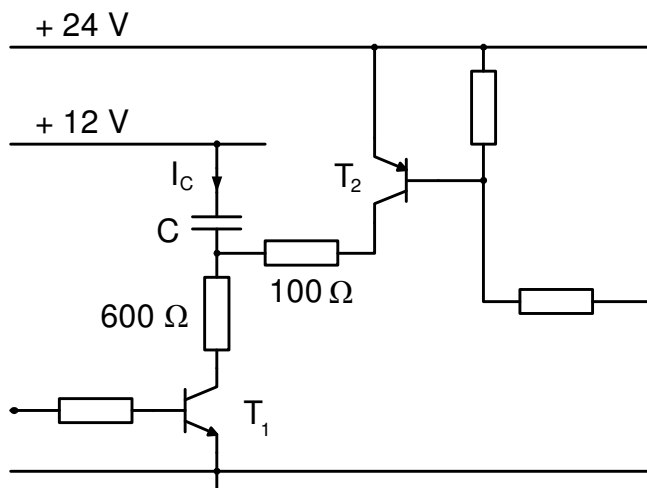
11. Przedstawiony czwórnik ma następujące własności selektywne:

- a. pasmowo – przepustowe
- b. stałe wzmocnienie niezależne od częstotliwości
- c. **pasmowo - zaporowe**
- d. dolnoprzepustowe



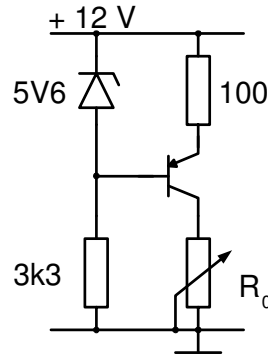
12. W układzie przedstawionym na rysunku załączamy idealny klucz T_1 , a po ustabilizowaniu się prądów i napięć równocześnie wyłączamy T_1 i załączamy idealny klucz T_2 . Określ różnicę pomiędzy ekstremalnymi wartościami prądu I_C :

- a. 200 mA
- b. **260 mA**
- c. 360 mA
- d. 480 mA



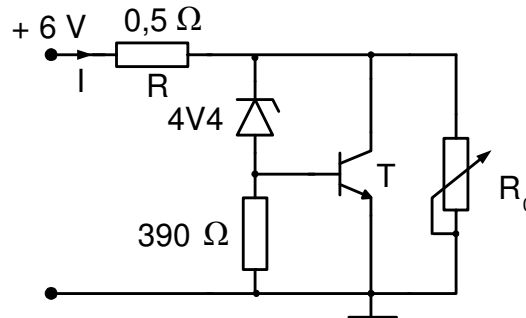
13. Dla źródła prądu określ dopuszczalny zakres zmian R_0 .

- a. $0 \div 0,14 \text{ k}\Omega$
- b. $0 \div 3,3 \text{ k}\Omega$
- c. $0 \div 12 \text{ k}\Omega$
- d. $0 \div 100 \text{ k}\Omega$



14. Dla równoległego stabilizatora napięcia 5V/dz (na rysunku) podaj wartość minimalną R_0 (pomijamy prąd dzielnika z diodą Zenera).

- a. 6Ω
- b. 3Ω
- c. $4,4\Omega$
- d. $2,5\Omega$

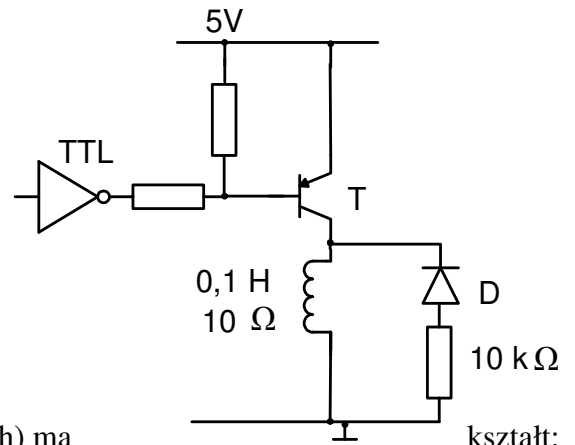


15. Jaka maksymalna moc wydziela się na tranzystorze regulującym T?

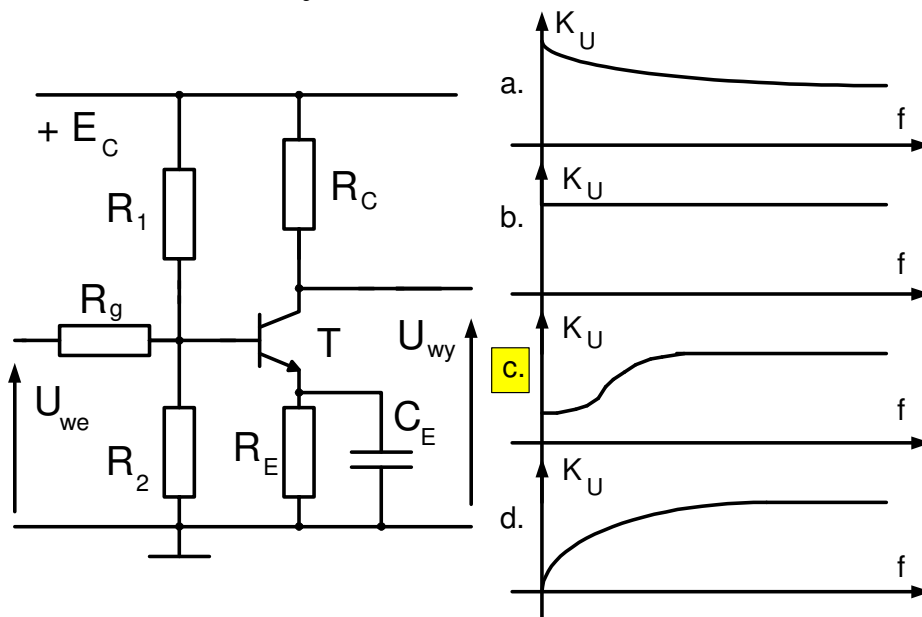
- a. 20W
- b. 13W
- c. 10W
- d. 5W

16. Dla poniższego układu, podać wartość minimalnego napięcia U_{CE} tranzystora prostującego T pracującego jako klucz (dioda D idealna).

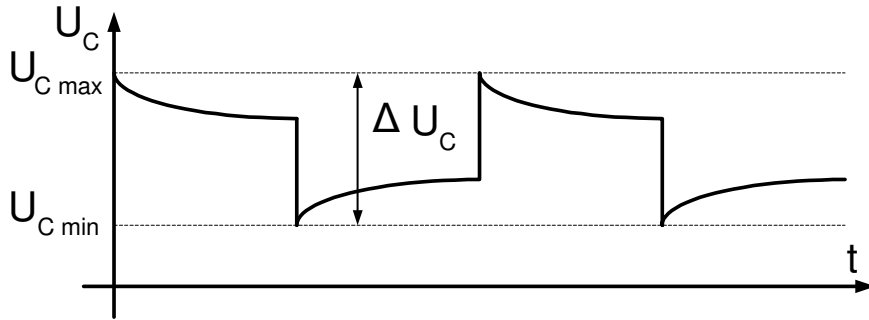
- a. $\approx 5\text{kV}$
- b. $\approx 500\text{V}$
- c. $\approx 50\text{V}$
- d. 5V



17. Charakterystyka wzmacniacza w funkcji częstotliwości (niskich) ma



18. Na wejście poniższego wzmacniacza podano przebieg prostokątny niskiej częstotliwości nałożony na dodatnią składową stałą, tak aby wzmacniacz pozostawał w obszarze aktywnym pomiędzy $V_{C \min}$ a $V_{C \max}$. Następnie na oscyloskopie zaobserwowano przebieg V_C .

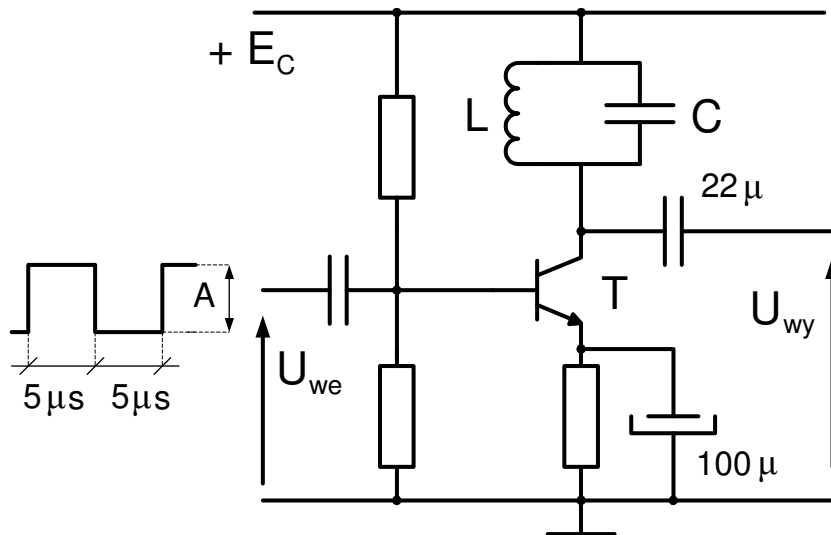


Aby uzyskać kształt prostokąta (amplituda ΔV_C nie musi zostać zachowana) należy:

- a. zwiększyć C_E 10- krotnie
- b. odłączyć C_E
- c. zwiększyć E_E
- d. Zwiększyć R_g

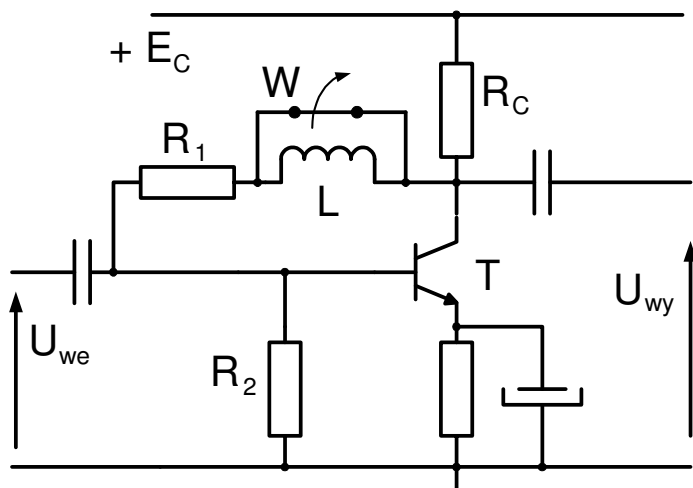
19. Na wejście wzmacniacza selektywnego o częstotliwości 400 kHz podano przebieg prostokątny jak na rysunku. Punkt pracy tranzystora dobrano tak, aby przy U_{we} jak na rysunku, w pełni wykorzystać napięcie zasilania E_C . Elementy L , T idealne. Amplituda przebiegu na wyjściu wyniesie:

- a. E_C
- b. $\frac{1}{2} E_C$
- c. $\frac{1}{4} E_C$
- d. 0



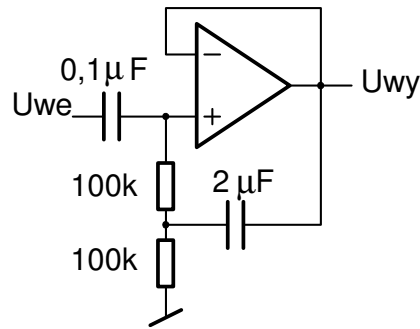
20. W poniższym wzmacniaczu punkt pracy dobrano optymalnie. Zakładamy, że T jest idealny a rezystancję cewki można pominąć. Po rozwarciu W uzyskamy:

- a. osłabienie n.cz.
- b. uwydatnienie w.cz.
- c. uwydatnienie n.cz.
- d. osłabienie w.cz.



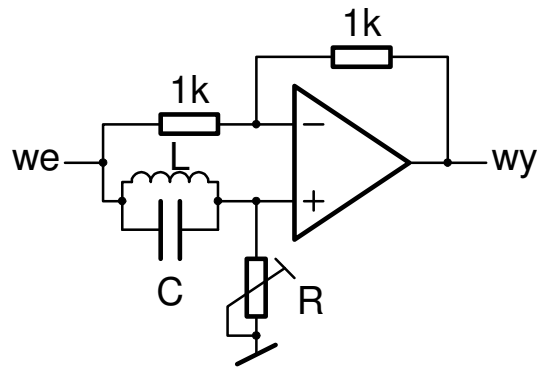
21. Wzmocnienie poniższego układu dla zakresu średnich częstotliwości wynosi około:

- a. 0,5
- b. **1**
- c. 2
- d. 3



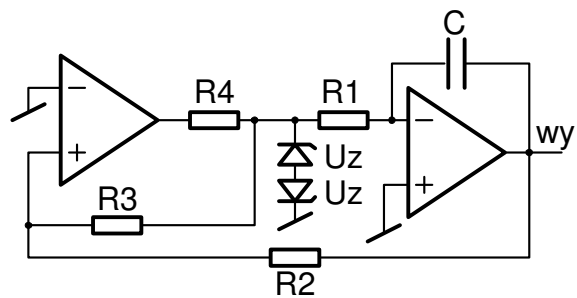
22. Wzmocnienie poniższego układu zależy:

- a. wprost od częstotliwości
- b. odwrotnie od częstotliwości
- c. **jest stałe w funkcji częstotliwości**
- d. w układzie dochodzi do generacji drgań



23. W poniższym układzie na wyjściu otrzymujemy:

- a. falę prostokątną
- b. sinusoidę
- c. obciętą sinusoidę
- d. **przebieg trójkątny**



24. W układzie z poprzedniego zadania wartość napięć maksymalnych (dodatnich i ujemnych) wynosi:

- a. $(U_z + 0,6V) \frac{R_3}{R_2 + R_4}$
- b. **$(U_z + 0,6V) \frac{R_2}{R_3}$**
- c. $(U_z + 0,6V) \frac{R_2}{R_3 + R_4}$
- d. $(U_z + 0,6V) \frac{R_3}{R_1}$

25. W układzie z poprzedniego zadania okres generowanego przebiegu wyniesie:

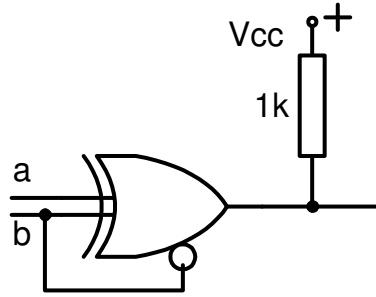
- a. $\frac{4R_3R_1C}{R_2}$
- b. $\frac{4R_2R_3C}{R_1}$
- c. **$\frac{4R_1R_2C}{R_3}$**
- d. $\frac{4R_1C}{R_2R_3}$

26. Jeżeli zachodzi: $f_1 \vee f_2 \vee f_3 = 1$, to prawdziwa jest zależność (f – funkcje Boole'a):

- a. $f_2 = \overline{f_1 \bullet f_3}$
- b. $f_2 = f_1 \oplus f_3$
- c. **$f_2 = \overline{f_1 \vee f_3}$**
- d. $f_2 = \overline{f_1 \bullet f_3}$

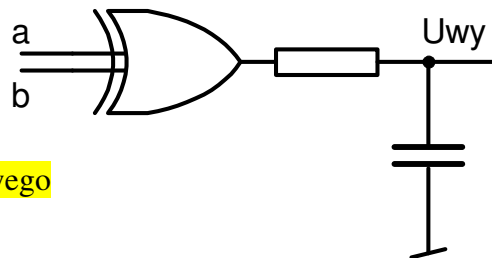
27. Bramkę sumy modulo 2 z wyjściem trójstanowym połączono jak poniżej. Jaka funkcja jest realizowana? (bramka aktywna, gdy na wejściu sterującym mamy poziom L)

- a. $\overline{a \oplus b}$
- b. $a \vee b$
- c. ab
- d. \overline{ab}



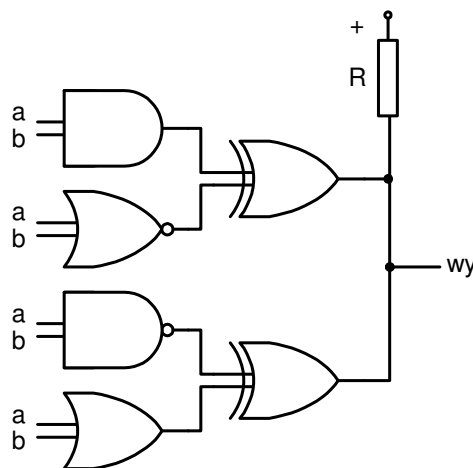
28. Poniższy układ może być wykorzystywany do: (a i b przebiegi periodyczne)

- a. pomiaru stosunku częstotliwości
- b. pomiaru średniego współczynnika wypełnienia
- c. pomiaru częstotliwości średniej
- d. pomiaru wzajemnego przesunięcia fazowego



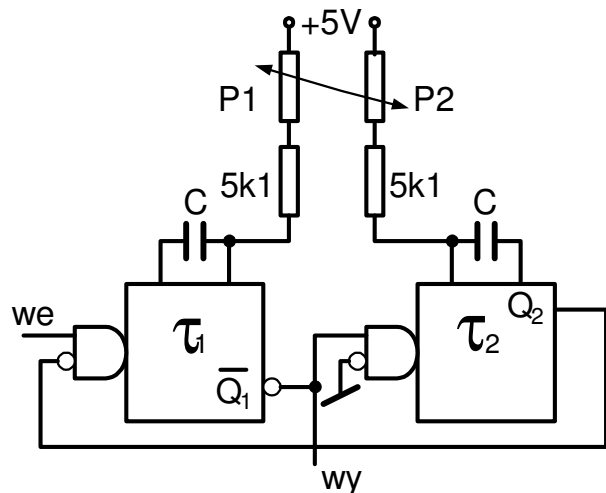
29. Dla poniższego połączenia funktorów logicznych uzyskamy na wyjściu dwóch bramek O.C. typu sumy modulo-dwa następującą funkcję:

- a. \overline{ab}
- b. \overline{ab}
- c. $\overline{a \oplus b}$
- d. 1



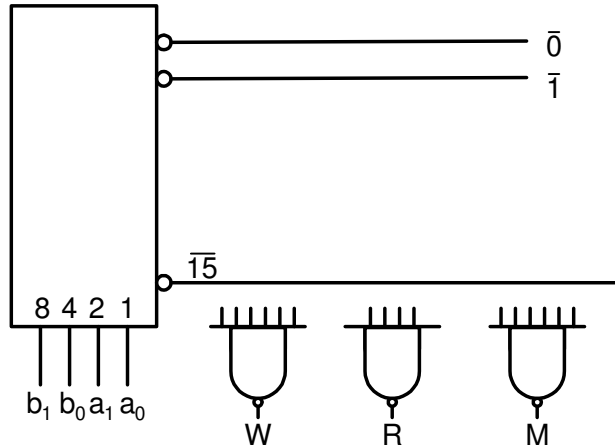
30. Poniższy układ może służyć

- a. jako generator przebiegów o regulowanym współczynniku wypełnienia ($w_e=1$)
- b. do wykrywania impulsów jedynkowych „ τ ” spełniających relację $\tau_1 < \tau < \tau_2$
- c. jw. dla relacji $\tau < \tau_1$ lub $\tau > \tau_2$
- d. do generacji wąskich impulsów po każdym zboczu impulsu τ na wejściu

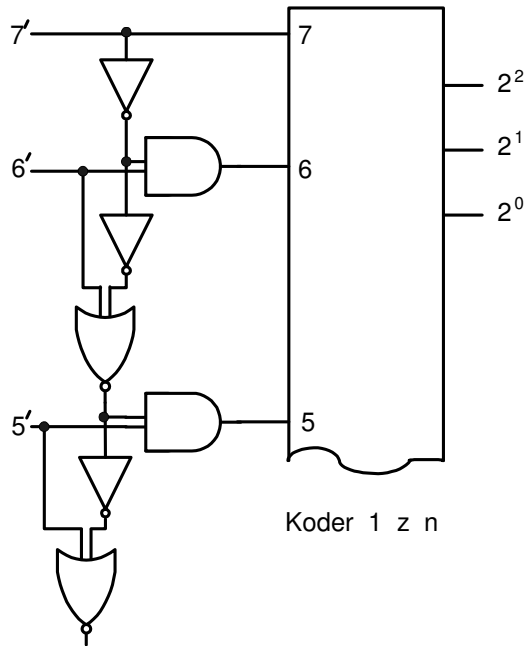


31. Aby zrealizować komparator równoległy dwóch 2-bitowych liczb binarnych $b_1 b_0 \div a_1 a_0$ zastosowano dekoder 4/16. Do generacji relacji M-R-W (mniejszy- równy- większy) wykorzystano trzy bramki NAND o odpowiedniej ilości wejść. Do ich wejść podamy wprost następujące zanegowane wyjścia dekodera:

	W	R	M
a.	$\overline{4}, \overline{8}, \overline{9}, \overline{12}, \overline{13}, \overline{14}$	$\overline{0}, \overline{5}, \overline{10}, \overline{15}$	$\overline{1}, \overline{2}, \overline{3}, \overline{6}, \overline{7}, \overline{11}$
b.	$\overline{1}, \overline{3}, \overline{6}, \overline{9}, \overline{11}, \overline{14}$	$\overline{0}, \overline{4}, \overline{8}, \overline{12}$	$\overline{2}, \overline{5}, \overline{7}, \overline{10}, \overline{13}, \overline{15}$
c.	$\overline{0}, \overline{4}, \overline{5}, \overline{11}, \overline{12}, \overline{15}$	$\overline{2}, \overline{6}, \overline{10}, \overline{14}$	$\overline{1}, \overline{3}, \overline{7}, \overline{8}, \overline{9}, \overline{13}$
d.	$\overline{2}, \overline{4}, \overline{5}, \overline{10}, \overline{11}, \overline{14}$	$\overline{3}, \overline{6}, \overline{9}, \overline{15}$	$\overline{0}, \overline{1}, \overline{7}, \overline{8}, \overline{12}, \overline{13}$

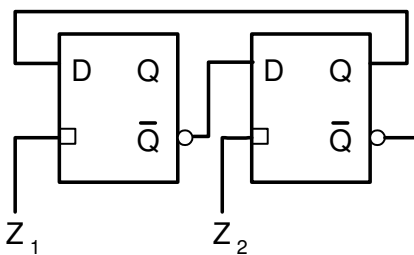


32. Na wejściu kodera kodu 1 z „n” na 3-bitowy kod binarny dokonano połączeń jak na rysunku. Uzyskujemy w ten sposób na wyjściach

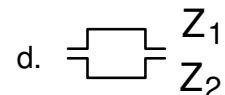


- a. kod Graya
- b. kod priorytetowy
- c. kod Johnsona
- d. kod linijki świetlnej

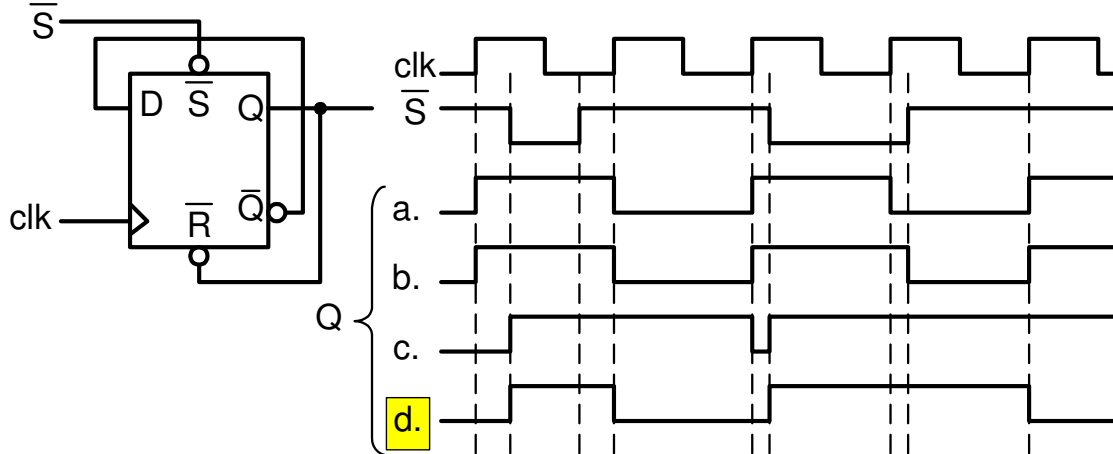
33. Aby uzyskać funkcje dwójki liczącej (zmiana stanu na przeciwny co okres zegarów Z_1, Z_2) wykorzystano dwa przerzutniki typu zatrzask. Jaki kształt zegarów Z_1, Z_2 jest wymagany?



a. nie jest możliwe

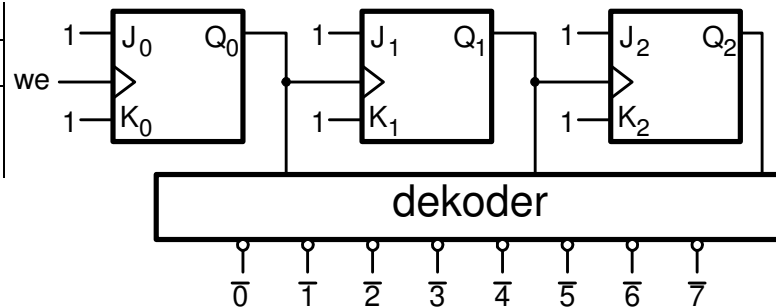


34. Przerzutnik D połączono jak na rysunku. Dla danych przebiegów clk i \bar{S} określ przebiegi na wyjściu Q



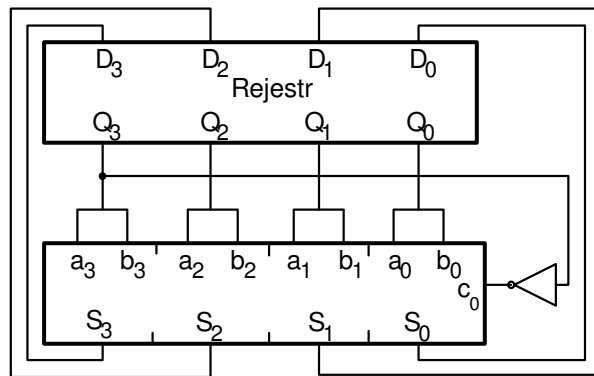
35. Licznik podłączono do dekodera zgodnie z rysunkiem. Do wyjść: $\bar{0}$, $\bar{4}$, $\bar{5}$ podłączono identyczny licznik. Zlicza on z krotnością „ $\frac{n}{8}$ ” w stosunku do pierwszego licznika:

	a.	b.	c.	d.
$\bar{0}$	3	1	2	2
$\bar{4}$	2	1	1	2
$\bar{5}$	1	1	3	1



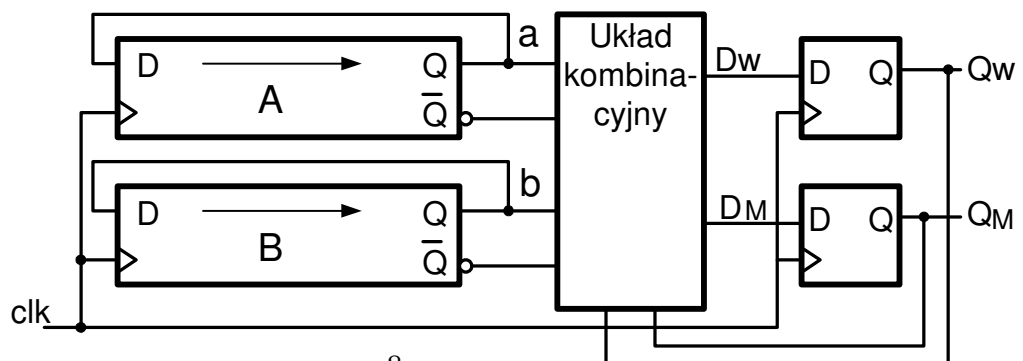
36. Łącząc rejestr z sumatorem, jak poniżej uzyskujemy n- stanowy licznik (wyjścia: $Q_3 Q_2 Q_1 Q_0$). Wartość N wynosi:

- a. 8
- b. 7
- c. 6
- d. 4



37. Komparator szeregowy porównuje dwie liczby A÷B począwszy od bitów najmłodszych. Bieżący stan porównywania zapamiętywany jest przez dwa przerzutniki: „W”- większy i „M” mniejszy (możliwe stany 10, 00, 01). Określ funkcje logiczne na wejściu Dw (przerzutnik „większy”). Przed porównaniem oba przerzutniki są zerowane.

- a. $(a \oplus \bar{b})Q_w$
- b. $a\bar{b} \vee Q_w(a \vee \bar{b})$
- c. $(a \oplus Q_w)\bar{b}$
- d. $a \vee \bar{b} \vee Q_w.a\bar{b}$



38. Jak wyżej tylko dla wejścia D_M

- a. $\bar{a} \vee b \vee Q_M \bar{a}b$ b. $(\bar{a} \oplus b)Q_M$ **c. $\bar{a}b \vee Q_M(\bar{a} \vee b)$** d. $(\bar{a} \oplus Q_M)b$

39. Interfejs szeregowy, asynchroniczny, odbiera ramkę formatu 8-bitowego z bitem parzystości (przebieg poniżej). Jaki bajt zostanie zidentyfikowany?

- a. 75h **b. AEh** c. 3Ah d. 57h



40. Jeśli odbiornik w interfejsie szeregowym stwierdzi tzw. błąd ramki, świadczy to o:

- a. przeciwnym stanie bitu parzystości
b. odebraniem bajcie spoza kodu ASCII
c. niedopasowaniu odbiornika do prędkości nadawcy
d. zbyt długim odstępem pomiędzy kolejnymi ramkami

41. Rozkaz korekcji dziesiętnej bajtu używamy w celu:

- a. konwersji liczby binarnej na dziesiętną
b. po rozkazie dodawania liczb w kodzie ASCII
c. w celu uzyskania wartości (100(dec)- liczba)- kod BCD
d. po rozkazie dodawania liczb w kodzie BCD

42. Mikroprocesor 8-bitowy wykonał odejmowanie liczb (7A-BE) hex. Uzyskany rezultat i stany bitów warunkowych (C- przeniesienie / *pożyczka*), Z- zerowość, N- znak, V- przekroczenie zakresu) wynoszą:

		V	N	Z	C
a.	BC hex	1	1	0	1
b.	38 hex	0	1	0	0
c.	44 hex	1	0	1	0
d.	6D hex	0	0	0	1

43. Uzyskany powyżej rezultat jest poprawny (+) lub niepoprawny (-) dla kodów NB (naturalny binarny) oraz U2 (uzupełnień do dwóch).

	NB	U2
a.	+	+
b.	+	-
c.	-	+
d.	-	-

44. W którym typie rozkazów musimy zwracać uwagę na kod używany do obliczeń (NB lub U2)?

- a. Przesunięcia** b. Dodawania c. Odejmowania d. Porównania

45. Mikrokontroler 8051 zaliczamy do układów typu:

- a. Harvard **b. von Neumana** c. pseudo- Harvard d. innych

46. Przenoszalność oprogramowania w przód w ramach tej samej rodziny μp (mikroprocesorów) nie jest możliwa, jeżeli nowy typ ma:

- a. nowe linie przerywające
- b. nowe rozkazy
- c. inną interpretację bitów warunkowych
- d. nowe rejestry

47. Przenoszalność oprogramowania wstecz w ramach tej samej rodziny μp (wykorzystujemy wyjątek nielegalnej instrukcji i emulację softwarową) jest niemożliwe, gdy nowszy μp posiada:

- a. nowe rejestry
- b. nowe tryby adresowania
- c. szybszy zegar
- d. nowe formaty danych

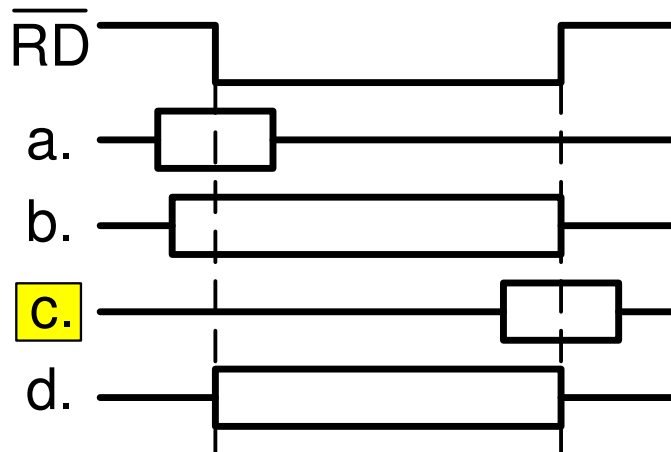
48. Instrukcje: DJNZ (dekrementacja i skok gdy nie zero) oraz CJNE (porównanie i skok gdy nierówne) można używać do organizacji pętli programowych. Indeks pętli może być zwiększany (+) bądź zmniejszany (-). Określ typowe zastosowanie obu rozkazów w każdej parze.

	DJNZ	CJNE
a.	-	-
b.	-	+
c.	+	-
d.	+	+

49. Ciąg programu w hipotetycznym μp zatrzymał się na instrukcji ALA: JMP ALA. Układ zaakceptował przerwanie. Aby powrót z przerwania nastąpił do następnej instrukcji, z pominięciem „zapętlonego” skoku, procedura przerywająca powinna:

- a. nie ingerować- działanie powyższe jest właściwe dla każdego μp
- b. zmodyfikować rejestr SP (wskaźnik stosu)
- c. przepisać stan SP do PC (licznik programu)
- d. zmodyfikować na stosie adres „ALA”

50. W czasie cyklu magistrali typu odczyt, hipotetyczny μp generuje strob odczytu $\overline{\text{RD}}$. Kiedy karta ma wysłać do μp ważne dane?



Etap B – Zadania

Technika analogowa

Problematyka ćwiczenia:

Pomiędzy urządzeniem nadawczym oraz odbiorczym przesyłany jest sygnał użyteczny w paśmie 10Hz ÷ 50kHz.

W trakcie odbioru sygnału po stronie odbiorczej stwierdzono obecność sygnału zakłócającego o częstotliwości 1kHz.

Temat ćwiczenia:

Zaprojektuj, wykonaj oraz zbadaj układ, który będzie eliminował sygnał o częstotliwości zakłócającej.

Układ powinien spełniać następujące wymagania:

- Wzmocnienie układu nie powinno przekraczać 6dB.
- Dobroć układu powinna być maksymalna do uzyskania w danej aplikacji.
- Rzeczywista częstotliwość środkowa f_0 powinna zawierać się w przedziale ($f_0 \pm 10\%$).

Dokumentacja dołączona do projektu powinna zawierać:

- Schemat ideowy proponowanego układu.
- Wszystkie obliczenia dotyczące przyjętych wartości elementów.
- Schemat układu pomiarowego do wyznaczenia charakterystyki amplitudowo- częstotliwościowej w zakresie od 10Hz do 50kHz .
- Tabelę wraz z proponowanymi częstotliwościami pomiarowymi.
- Charakterystykę amplitudowo-częstotliwościową wykreśloną na papierze milimetrowym. Na charakterystyce amplitudowo-częstotliwościowej proszę graficznie wyznaczyć dobroć układu.

Uwaga!

- ✓ **Elementy układu powinny być dobrane z zestawu dołączonego do ćwiczenia.**
- ✓ **Wszystkie niezbędne informacje konieczne do wykonania wybranego (jednego z dwóch) układu znajdziesz na następnych stronach.**

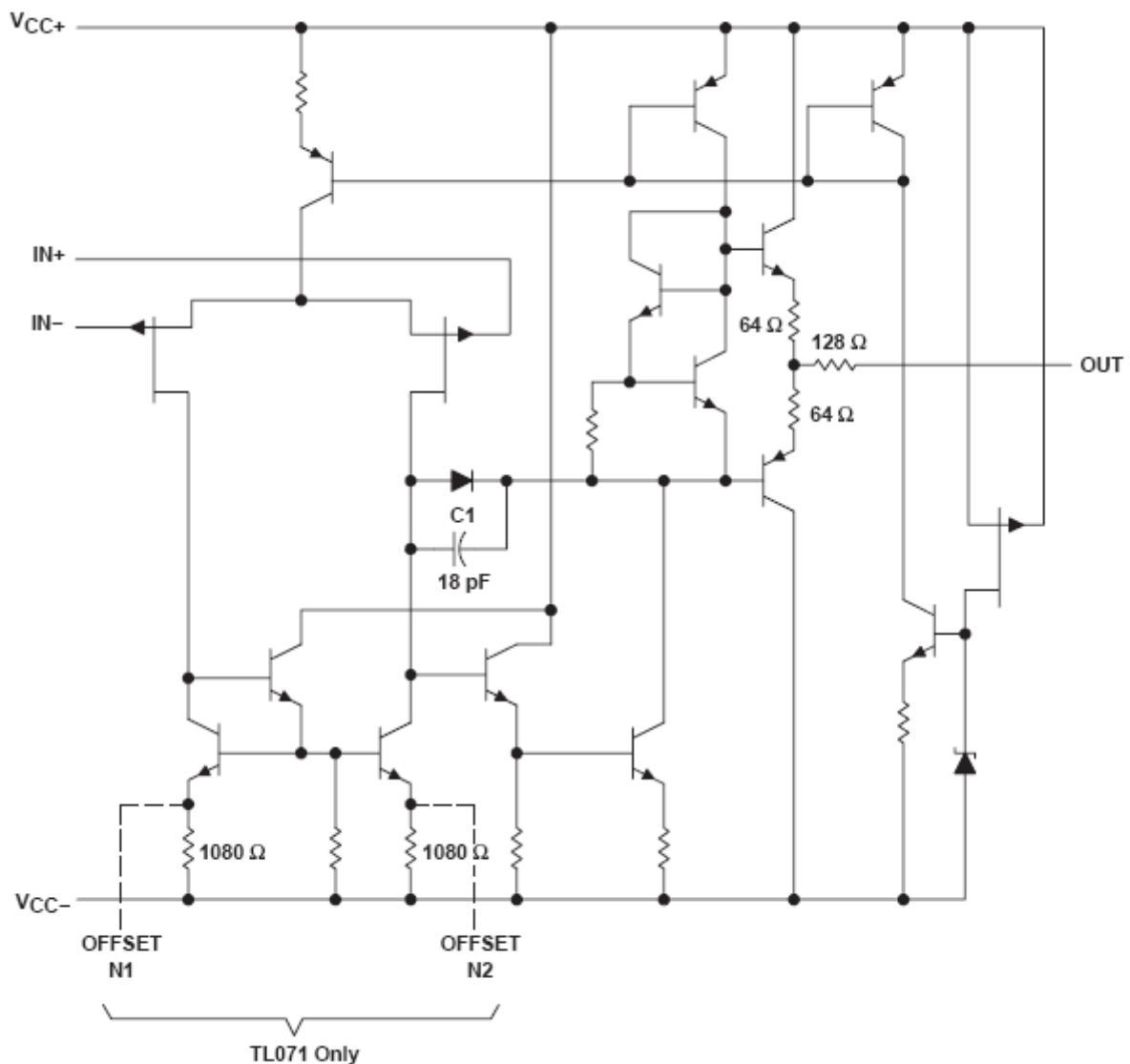
Zastosowanie układu TL 071 w aplikacjach filtrów pasmowo – zaporowych.

1. Obudowa i wyprowadzenia – widok z góry oraz schemat aplikacyjny.



Układ TL 071 jest wzmacniaczem operacyjnym wykonanym w technologii JFET. Charakteryzuje się niskim poziomem zniekształceń nieliniowych, niskimi szumami własnymi, a także posiada wewnętrzny układ kompensacji częstotliwości. Zalety te pozwalają na szerokie zastosowania układu w technice akustycznej.

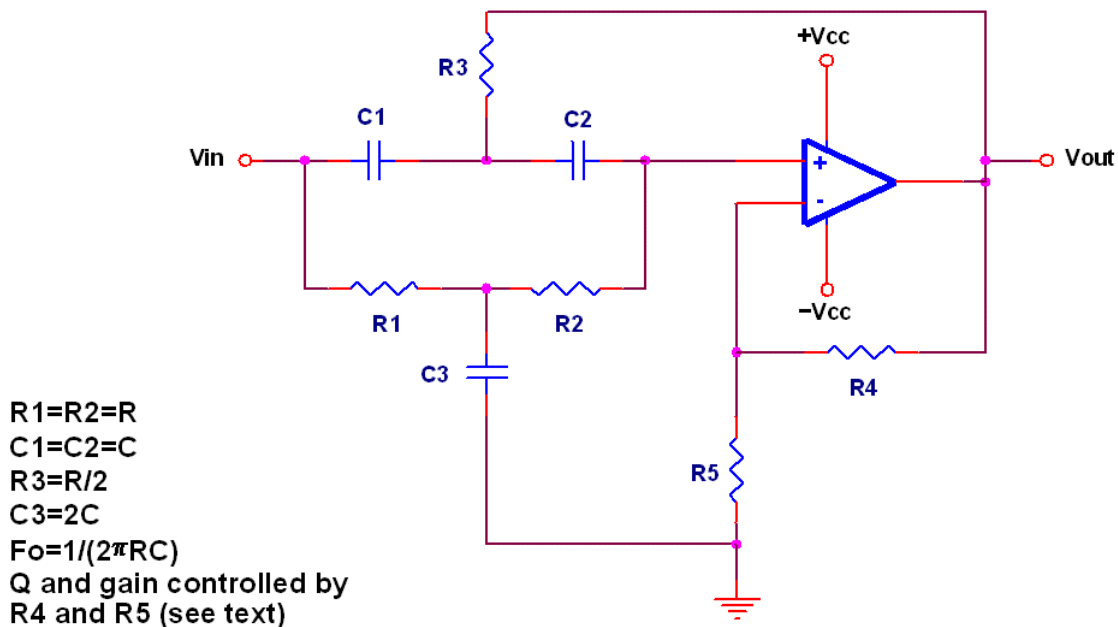
2. Struktura wewnętrzna układu TL 071.



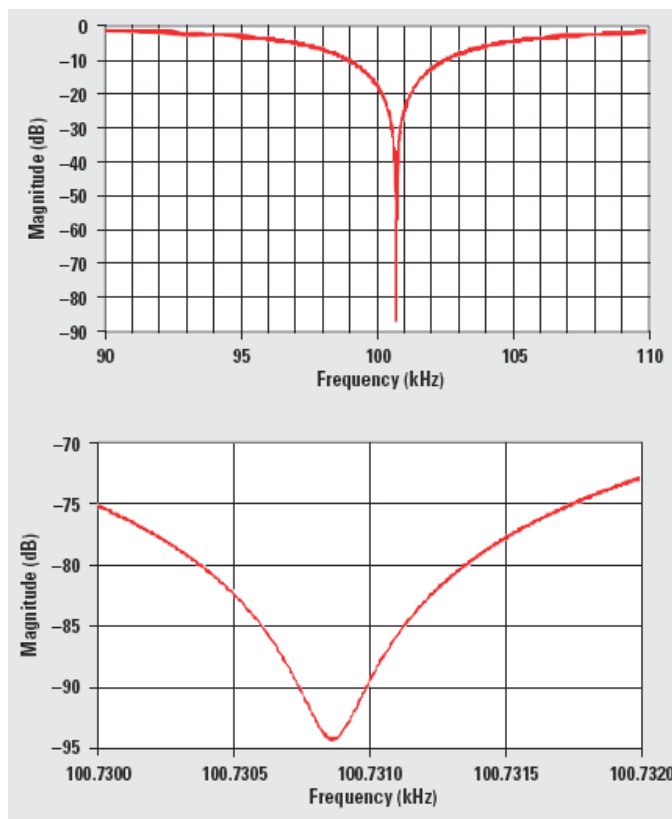
3. Zastosowanie TL 071 do budowy filtrów pasmowo – zaporowych w strukturze Sallen – Key.

- Układ filtra pasmowo – zaporowego z jednym wzmacniaczem operacyjnym (Twin – T notch filter).

- Schemat ideowy układu filtra.



- Charakterystyka amplitudowo – częstotliwościowa filtra.



- **Opis układu filtra.**

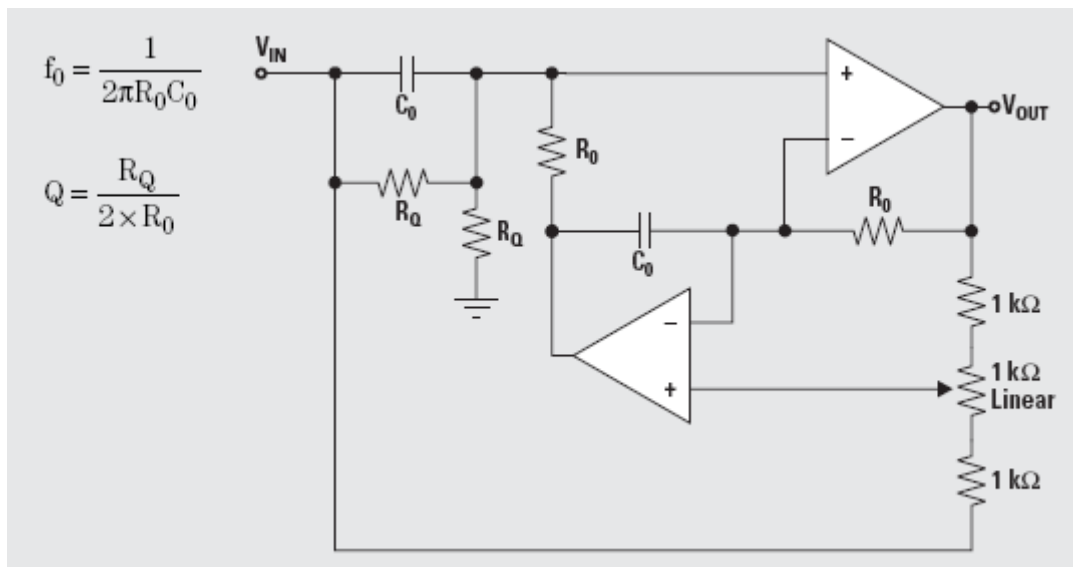
Struktury Sallen–Key są używane do realizacji filtrów pasmowo–zaporowych o bardzo wysokiej dobroci Q. Realizacja możliwa jest z zastosowaniem pojedynczego wzmacniacza operacyjnego serii TL 071. Dla poprawnej pracy układu wymagana jest duża dokładność w doborze obliczonych wartości elementów, z zachowaniem tolerancji rzędu 1 %. Najbardziej stromą charakterystykę filtra uzyskujemy przy większej wartości dobroci, kiedy zastosowane elementy spełniają następujące warunki:

$$\begin{aligned} R_1 &= R_2 = R \\ R_3 &= R/2 \\ C_1 &= C_2 = C \\ C_3 &= 2 * C \end{aligned}$$

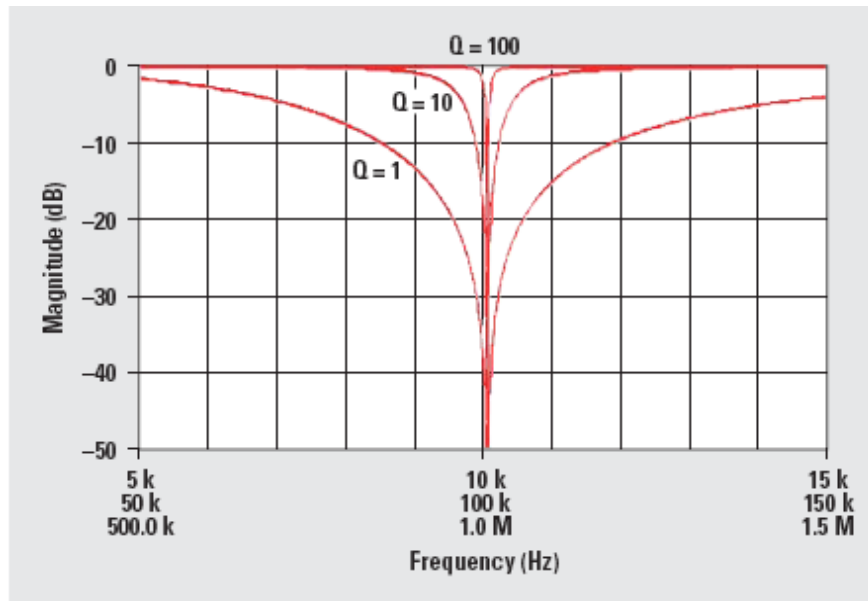
Istnieje możliwość dokładniejszego dostrojenia środkowej częstotliwości filtra poprzez zastosowanie dodatkowego potencjometru nastawnego, szeregowo z rezystorem R_3 . Rezystory R_4 i R_5 służą do kontroli wzmocnienia i dobroci. Rezystor R_4 nie powinien być równy R_5 , w przeciwnym razie układ nie będzie pracował stabilnie. W praktyce R_5 powinien posiadać większą (maksymalnie 2 razy) wartość od R_4 , co pozwoli na ograniczenie wzmocnienia układu do poziomu poniżej 6 dB.

- **Układ filtra pasmowo – zaporowego z dwoma wzmacniaczami operacyjnymi (Fliege notch filter).**

- **Schemat ideowy układu filtra oraz podstawowe zależności stosowane w obliczeniach.**



- **Charakterystyka amplitudowo – częstotliwościowa filtra.**



- **Opis układu filtra.**

Ten rodzaj filtra z rodziny Sallen – Key posiada kilka zalet w porównaniu z układem poprzednim:

- Częstotliwość środkowa filtra jest uzyskiwana poprzez odpowiedni dobór elementów R_0 , C_0 , dla zadanej częstotliwości, przy wykorzystaniu przedstawionych powyżej zależności.
- Dla przyjętej wartości dobroci Q , obliczamy wartość rezystora R_Q . Aby układ działał poprawnie należy zastosować elementy o wartościach zbliżonych do uzyskanych na drodze obliczeń wartości. Zaleca się zachowanie tolerancji 1%.
- Niedokładność zastosowanych elementów spowoduje zmianę częstotliwości środkowej f_0 filtra, nie zaś jego dobroci.
- Dobroć filtra może być regulowana w pewnym zakresie niezależnie od wartości częstotliwości środkowej f_0 , poprzez zastosowanie dwóch rezystorów 1 k Ω oraz liniowego potencjometru nastawnego 1 k Ω . W tym przypadku nie jest wymagany dobór wartości elementów z zachowaniem tolerancji 1 %.

Wadą powyższego układu jest zastosowanie dwóch wzmacniaczy operacyjnych oraz większa złożoność.



XXX Ogólnopolska Olimpiada Wiedzy
Elektrycznej i Elektronicznej
3 - 4 kwietnia 2006 r.
Zespół Szkół Łączności w Krakowie

Zawody II stopnia etap II - ćwiczenia laboratoryjne

Grupa Elektroniczna

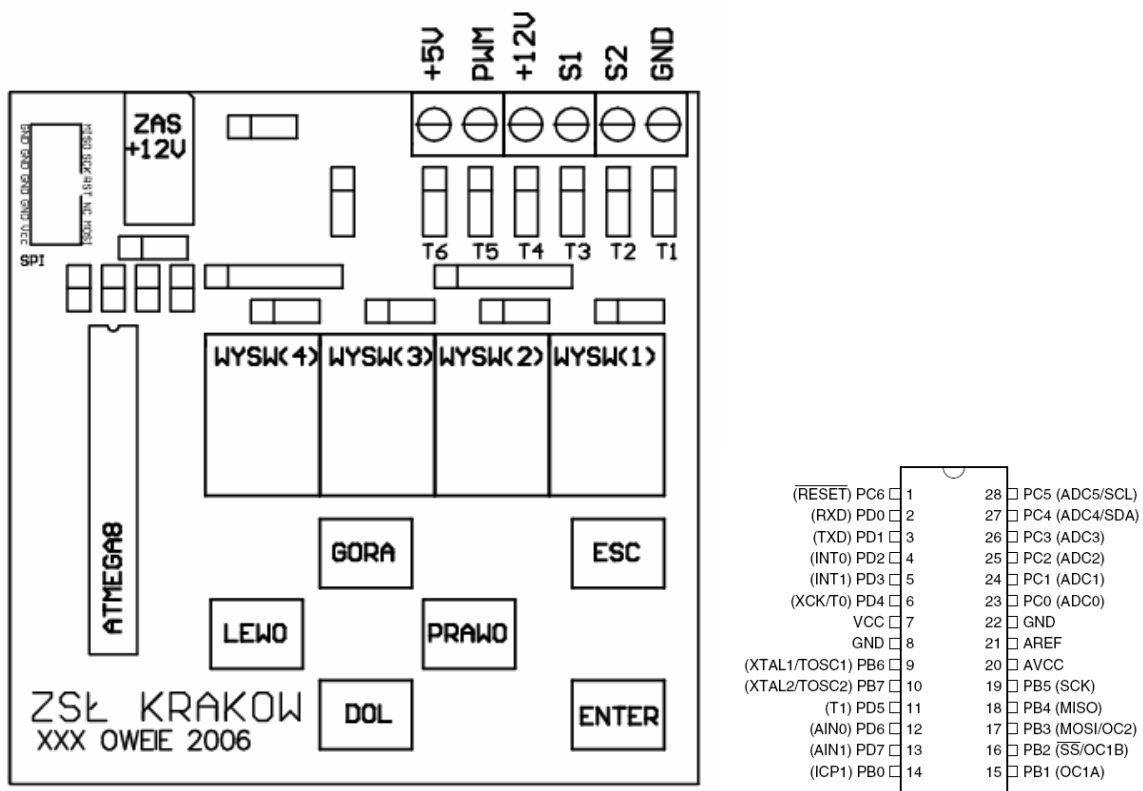
Sterownik natężenia oświetlenia

Cel:

Celem ćwiczenia jest opracowanie programu pozwalającego na sterowanie natężeniem świecenia żarówki o mocy 3W i znamionowym napięciu pracy 14V. Konstrukcja sterownika jest oparta na mikrokontrolerze jednocukłowym ATMEGA8 wyposażonym w interfejsy: multipleksowanego wyświetlacza siedmiosegmentowego LED, 6 przyciskowej klawiatury oraz układu wykonawczego z tranzystorem mocy NMOS.

Opis układu

Głównym elementem sterownika jest 8 bitowy mikrokontroler jednocukłowy o architekturze RISC. Pracuje on z wewnętrznym oscylatorem o częstotliwości ok. 1MHz. Do prezentacji informacji zastosowano 2 podwójne wyświetlacze siedmiosegmentowe o wspólnej anodzie, które są sterowane przez tranzystory pnp BD136, natomiast wszystkie katody odpowiednich segmentów są połączone razem i przez rezystory ograniczające są bezpośrednio dołączone do wyjść procesora. Układem wykonawczym w modelu jest tranzystor mocy NMOS (IRF520), którego bramka jest bezpośrednio dołączona do wyjścia modulatora PWM, wbudowanego w kontroler.



Rys.1. Widok płytki sterownika oraz wyprowadzenia procesora.

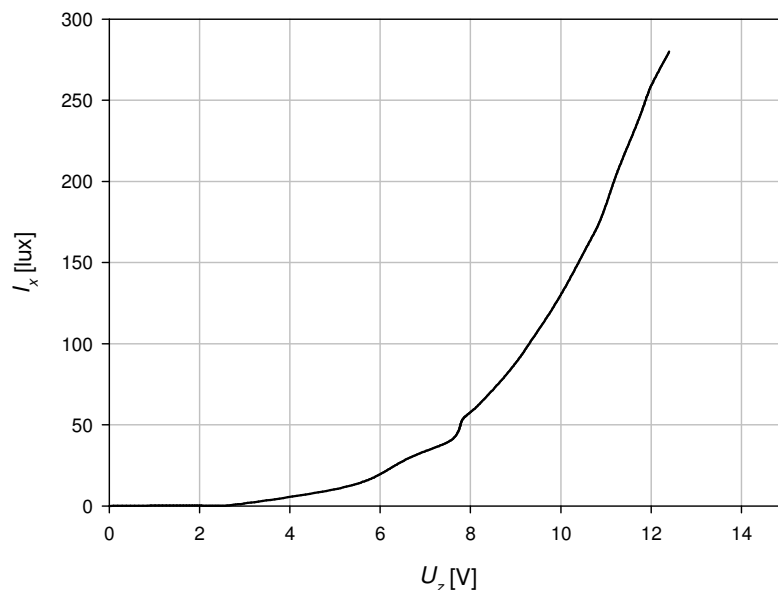
Poniżej przedstawiono zestawienie wyprowadzeń procesora wraz z opisem funkcji pełnionych w sterowniku.

Wyświetlacz 7 segmentowy LED : katody segmentów			
Zapalenie segment wymaga ustawienia stanu „0” na odpowiednim porcie procesora.			
Oznaczenie w programie	Port procesora	Numer	Funkcja
Port_seg_a	Portc.5	PC.5 (28)	segment a
Port_seg_b	Portc.4	PC.4 (27)	segment b
Port_seg_c	Portc.3	PC.3 (26)	segment c
Port_seg_d	Portc.2	PC.2 (25)	segment d
Port_seg_e	Portc.1	PC.1 (24)	segment e
Port_seg_f	Portb.5	PB.5 (19)	segment f
Port_seg_g	Portb.4	PB.4 (18)	segment g
Port_seg_p	Portb.3	PB.3 (17)	przecinek
Wyświetlacz 7 segmentowy LED : anody wyświetlaczy,			
Wybór wyświetlacza wymaga ustawienia stanu „0” na odpowiednim porcie.			
Port_seg4	Portd.3	PD.3(5)	Wybór segmentu 4 (ostatni z lewej)
Port_seg3	Portd.2	PD.2(4)	Wybór segmentu 3
Port_seg2	Portd.1	PD.1(3)	Wybór segmentu 2
Port_seg1	Portd.0	PD.0(2)	Wybór segmentu 1 (ostatni z prawej)
Klawisze : przyciśnięcie klawisza ściąga wejście kontrolera do masy			
Klaw_gora	Pinb.7	PB.7(10)	Klawisz strzałka do góry
Klaw_dol	Pinb.0	PB.0(14)	Klawisz strzałka do dołu
Klaw_lewo	Pinb.6	PB.6(9)	Klawisz strzałka w lewo
Klaw_prawo	Pind.5	PD.5(11)	Klawisz strzałka w prawo
Klaw_enter	Pind.7	PD.7(13)	Klawisz Enter
Klaw_esc	Pind.6	PD.6(12)	Klawisz ESC
PWM : wyjście modulatora PWM kontrolera			
PWM1A	OC1A	PB.1(15)	Wyjście układu <i>Output Compare 1A</i> licznika T1 w konfiguracji modulatora PWM

Tabela 1. Charakterystyka natężenia oświetlenia w funkcji napięcia na włóknie żarówki.

I_x [lux]	0	0,3	1	10,6	20,8	31,3	40,9	52,2	60,2	81,5	100	121	140	161
U_z [V]	0	2,0	2,85	5,06	6,08	6,78	7,59	7,8	8,1	8,81	9,3	9,8	10,2	10,6

I_x [lux]	178	202	223	244	259	280
U_z [V]	10,9	11,2	11,5	11,6	11,8	12,0

Rys.1. z. Charakterystyka $I_x = f(U_z)$ **Zadania:**

1. Wykaż, że w przedstawionym rozwiązaniu sterownika z modulatorem PWM można płynnie regulować natężenie światła emitowanego przez żarówkę, narysuj uproszczony schemat tego układu.
2. Narysuj diagram przepływu sterowania (algorytm) programu realizującego następujące wymagania:
 - a. liniową regulację natężenia oświetlenia, realizowaną przez mechanizm tablicy odniesień, w zakresie umownej skali od 0-19, przy czym:
 - i. 0 odpowiada wyłączeniu żarówki,
 - ii. 19 pełnej intensywności świecenia,
 - iii. regulacja natężania dokonywana jest za pomocą dwóch przycisków, z eliminacją procesu drgania zestyków,
 - b. prezentację bieżącej wartości umownego natężenia światła na dwóch ostatnich cyfrach wyświetlacza multipleksowanego w postaci 00, 01, ...19.
 - c. sprawdzanie granic zmiany umownego natężenia światła (po osiągnięciu wartości najmniejszej lub największej dalsza aktywność przycisków zmniejszania lub zwiększania jest ignorowana)

Uwaga: *Pomocą w realizacji zadania jest opis modelu oraz szkic programu pwm_zad.bas zawartego w Dodatku I.*

3. Uzupełnić zgodnie z zaproponowanym algorytmem szkic programu, dokonać jego kompilacji oraz załadować do procesora celem sprawdzenia poprawności rozwiązania. Mechanizm kompilacji oraz ładowania programu zawiera Dodatek II, natomiast dodatek III zawiera opis wszystkich niezbędnych instrukcji języka BASCOM potrzebnych do rozwiązania zadania, można także skorzystać i pliku pomocy w języku polskim znajdującym się w folderze Olimpiada.

Dodatek I. Listing programu

```
' XXX OWEIE Zadanie praktyczne - część cyfrowa
' Kraków 2006
' ***** Regulator oświetlenia *****
' UWAGI:
' 1. W kodzie programu znajduje się miejsce na rozwiązanie zadania,
'   w miejsca Xx należy wpisać właściwe wartości lub wyrażenia,
'   przedstawiony szkic rozwiązania można dowolnie zmieniać.
' 2. Środowisko programowe jest już właściwie skonfigurowane,
' 3. Należy skupić się jedynie na realizacji algorytmu wykorzystując
'   otrzymaną listę instrukcji
' *****
' Konfiguracja procesora : PROSZĘ NIE MODYFIKOWAĆ !!!
' *****
' Deklaracja typu procesora: ATMEGA8
$regfile = "m8DEF.DAT"
' Częstotliwość zegara wewnętrznego
' procesora ATMEGA8
$crystal = 1000000
' Określenie nazw portów wyjściowych:
' Wyjście do podłączenia żarówki
Zarowka Alias Portb.1
' Wyświetlacze 7 segmentowe o wspólnej
' anodzie
' Zapalenie wyświetlacza wymaga ustawienia
' 0 na odpowiednim porcie
Port_seg_a Alias Portc.5 ' segment a
Port_seg_b Alias Portc.4 ' segment b
Port_seg_c Alias Portc.3 ' segment c
Port_seg_d Alias Portc.2 ' segment d
Port_seg_e Alias Portc.1 ' segment e
Port_seg_f Alias Portb.5 ' segment f
Port_seg_g Alias Portb.4 ' segment g
Port_seg_p Alias Portb.3 ' przecinek
' Nazwy aliasowe pinów:
Pin_seg_a Alias Pinc.5 ' segment a
Pin_seg_b Alias Pinc.4 ' segment b
Pin_seg_c Alias Pinc.3 ' segment c
Pin_seg_d Alias Pinc.2 ' segment d
Pin_seg_e Alias Pinc.1 ' segment e
Pin_seg_f Alias Pinb.5 ' segment f
Pin_seg_g Alias Pinb.4 ' segment g
Pin_seg_p Alias Pinb.3 ' przecinek
' Anody wyświetlaczy, wybór wymaga
' ustawienia 0 na odpowiednim porcie
Port_seg4 Alias Portd.3
Port_seg3 Alias Portd.2
Port_seg2 Alias Portd.1
Port_seg1 Alias Portd.0
' Nazwy aliasowe pinów:
Pin_seg4 Alias Pind.3
Pin_seg3 Alias Pind.2
Pin_seg2 Alias Pind.1
Pin_seg1 Alias Pind.0
Pin_seg1 Alias Pind.0
' Określenie nazw wejść:
' Klawiatura :
Klaw_gora Alias Pinb.7
Klaw_dol Alias Pinb.0
Klaw_lewo Alias Pinb.6
Klaw_prawo Alias Pind.5
Klaw_enter Alias Pind.7
Klaw_esc Alias Pind.6
' Nazwy aliasowe wejść
' Klawiatura :
Port_klaw_gora Alias Portb.7
Port_klaw_dol Alias Portb.0
Port_klaw_lewo Alias Portb.6
Port_klaw_prawo Alias Portd.5
Port_klaw_enter Alias Portd.7
Port_klaw_esc Alias Portd.6
' Konfiguracja portów wyjściowych
' i wejściowych
' Wyjścia do sterowania wyświetlaczem LED
Config Pin_seg_a = Output
Config Pin_seg_b = Output
Config Pin_seg_c = Output
Config Pin_seg_d = Output
Config Pin_seg_e = Output
Config Pin_seg_f = Output
Config Pin_seg_g = Output
Config Pin_seg_p = Output
Config Pin_seg4 = Output
Config Pin_seg3 = Output
Config Pin_seg2 = Output
Config Pin_seg1 = Output
' Konfiguracja wejść
Config Klaw_gora = Input
Config Klaw_dol = Input
Config Klaw_lewo = Input
Config Klaw_prawo = Input
Config Klaw_enter = Input
Config Klaw_esc = Input
```

```
*****
```

```
'Konfiguracja układów czasowo/licznikowych
*****
' Timer1 w konfiguracji 8 bitowego generatora PWM
Config Timer1 = Pwm , Pwm = 8 , Compare A Pwm = Clear Down , Compare B Pwm = Clear Down ,
Prescale = 8
' Timer0 w konfiguracji czasomierza
' do odmierzenia czasu odświeżania wyświetlacza multipleksowanego
' odświeżanie całego wyświetlacza, co 24ms
Config Timer0 = Timer , Prescale = 1024
*****
'Deklaracja procedur, procedur obsługi przerwania oraz zmiennych
*****
'Procedura ustawiania współczynnika wypełnienia generatora PWM
Declare Sub Ustaw_pwm(byval Pw As Byte) 'ustawia określony współczynnik wypełnienia
' Wektor przerwania od Timer0
On Timer0 Irq_timer0
' Deklaracja zmiennych wyświetlacza LED
Dim Index_seg As Byte
Dim Wysw(4) As Byte
' *****
' Początek rozwiązywania zadania
' *****
' Deklaracja zmiennych:
' Umowny wskaźnik natężenia światła (przyjme zakres od 0 do 19)
Dim I As Byte
' Wartość wpisywana do układu generatora PWM – (determinuje współczynnik wypełnienia)
Dim W As Byte
' Bit potwierdzający odczyt z klawiatury nowych danych
Dim Newdata As Bit

*****
' Początek programu
*****
' Nadanie zmiennym wartości początkowej
Index_seg = 0 'index bieżącej pozycji wyświetlacza multipleksowanego
Call Ustaw_pwm(0) 'wartość początkowa generatora PWM (wyłączony)

' Odblokowanie systemu przerwania
Enable Interrupts
' Odmaskowanie przerwania od Timer0 (obsługa wyświetlania)
Enable Timer0

'Ustawienie rezystorów podciągających na wejściach klawiszy
Set Port_klaw_gora
Set Port_klaw_dol
'do zrobienia ...

'Wyświetlacz (ustawienia początkowe)
Wysw(1) = 0 'na pozycji 1 wyświetlacza cyfra 0, bo zmienna I też ma wartość 0

Wysw(2) = 0 'na pozycji 2 wyświetlacza cyfra 0, bo zmienna I też ma wartość 0
Wysw(3) = 10 'wyświetlacz wygaszony (cyfra spoza zakresu)
Wysw(4) = 10 'wyświetlacz wygaszony (cyfra spoza zakresu)

*****
'Pętla główna
```

```
'*****  
Do 'pętla nieskończona  
' Sprawdź czy Klaw_gora (strzałka w górę) jest wciśnięty  
If Klaw_gora = 0 Then  
  Waitms 50  
' Sprawdź czy klawisz jest stabilny  
If Klaw_gora = 0 Then  
' Jeżeli odczyt poprawny to zwiększ zmienną I o jeden  
  Incr I  
' ( pamiętaj o granicach zmiennej I max to 19)  
If I = 20 Then  
  I = 19  
End If  
' Potwierdź, że otrzymano nowe ważne dane  
  Newdata = 1  
End If  
End If  
  
While Klaw_gora = Xx  
  ' czekaj na zwolnienie klawisza  
Wend  
  
' to samo zrób z klawiszem Klaw_dol (strzałka w dół), ale tym razem zmniejsz wartość zmiennej I  
' do zrobienia ...  
  
  ' sprawdź granice, jeżeli zero to już nie zmniejszaj  
If I = 255 Then  
  I = 0  
End If  
  Newdata = 1  
End If  
End If  
  
  ' czekaj na zwolnienie klawisza  
  
  
'poczekaj 50ms  
Waitms 50  
  
' Jeżeli były nowe dane to wyświetl wartość zmiennej I oraz modyfikuj PWM  
If Newdata = 1 Then  
  If I > 9 Then  
    Wysw(2) = 1  
    Wysw(1) = I - 10  
  Else  
    Wysw(2) = 0  
    Wysw(1) = I  
  End If  
' Dokonaj konwersji indeksu I na wartość wpisywaną do generatora PWM  
Xx = Lookup(Xx , Tablica_korekcji_pwm)  
Call Ustaw_pwm(W)  
  Newdata = 0  
End If
```



```

Loop ' Koniec głównej pętli
End

' Tablica konwersji z kilkoma przykładowymi danymi, pomocą w jej uzupełnieniu jest opis procedury
' Ustaw_pwm(w) znajdujący się w Dodatek III
Tablica_korekcji_pwm:
  Data 0 , 42 , Xx , Xx , Xx , Xx , Xx , Xx , Xx , Xx , Xx , Xx , Xx , 231 , Xx , 244 , Xx , 250 ,
  255
*****
' KONIEC ROZWIĄZANIA
*****

*****
' Definicje procedur oraz obsługi przerwania
*****

'Obsługa przerwania od Timer0
Reset Port_seg_c
Irq_timer0:
  Reset Port_seg_d
  Tcnt0 = 254
  Reset Port_seg_g
' wartość początkowa zliczania licznika T0
Case 4 :
  ' Wyświetlanie nieaktywne
  Reset Port_seg_b
  Set Port_seg4
  Reset Port_seg_c
  Set Port_seg3
  Reset Port_seg_f
  Set Port_seg2
  Reset Port_seg_g
  Set Port_seg1
Case 5 :
  'Segmenty nieaktywne
  Reset Port_seg_a
  Set Port_seg_a
  Reset Port_seg_c
  Set Port_seg_b
  Reset Port_seg_d
  Set Port_seg_c
  Reset Port_seg_f
  Set Port_seg_d
  Reset Port_seg_g
  Set Port_seg_e
  Reset Port_seg_e
  Set Port_seg_f
  Reset Port_seg_a
  Set Port_seg_g
  Reset Port_seg_c
  Set Port_seg_d
  Reset Port_seg_d
  Reset Port_seg_e
  Reset Port_seg_f
  Set Port_seg_g
  Reset Port_seg_g

' Konwersja cyfr BCD na kody wyświetlacza
' 7 segmentowego
Reset Port_seg_f
Reset Port_seg_g
Select Case Wysw(index_seg)
Case 7 :
  Case 0 :
    Reset Port_seg_a
    Reset Port_seg_b
    Reset Port_seg_c
    Reset Port_seg_c
    Reset Port_seg_d
    Reset Port_seg_d
    Reset Port_seg_e
    Reset Port_seg_e
    Reset Port_seg_f
    Reset Port_seg_f
  Case 1 :
    Reset Port_seg_d
    Reset Port_seg_d
    Reset Port_seg_e
    Reset Port_seg_e
    Reset Port_seg_f
    Reset Port_seg_f
  Case 2 :
    Reset Port_seg_g
    Reset Port_seg_g
    Reset Port_seg_a
    Reset Port_seg_a
    Reset Port_seg_b
    Reset Port_seg_b
    Reset Port_seg_d
    Reset Port_seg_d
    Reset Port_seg_e
    Reset Port_seg_e
  Case 3 :
    Reset Port_seg_g
    Reset Port_seg_g
    Reset Port_seg_f
    Reset Port_seg_f
  Case 4 :
    Reset Port_seg_g
    Reset Port_seg_g
    Reset Port_seg_f
    Reset Port_seg_f
  Case 5 :
    Reset Port_seg_g
    Reset Port_seg_g
    Reset Port_seg_f
    Reset Port_seg_f
  Case 6 :
    Reset Port_seg_g
    Reset Port_seg_g
    Reset Port_seg_f
    Reset Port_seg_f
  Case 7 :
    Reset Port_seg_g
    Reset Port_seg_g
    Reset Port_seg_f
    Reset Port_seg_f
  Case 8 :
    Reset Port_seg_g
    Reset Port_seg_g
    Reset Port_seg_f
    Reset Port_seg_f
  Case 9 :
    Reset Port_seg_g
    Reset Port_seg_g
    Reset Port_seg_f
    Reset Port_seg_f
End Select

```

```
Select Case Index_seg
  Case 1 : Reset Port_seg1
  Case 2 : Reset Port_seg2
  Case 3 : Reset Port_seg3
  Case 4 : Reset Port_seg4
End Select

Incr Index_seg

' czy juz wszystkie wyświetlacze
If Index_seg = 5 Then
  Index_seg = 1
End If

Return

Sub Ustaw_pwm(byval Pw As Byte)
  Pwm1a = Pw
End Sub
```

Dodatek II

1. Kompilacja:

Aby dokonać kompilacji należy z menu Program wybrać opcję Compile (F7).

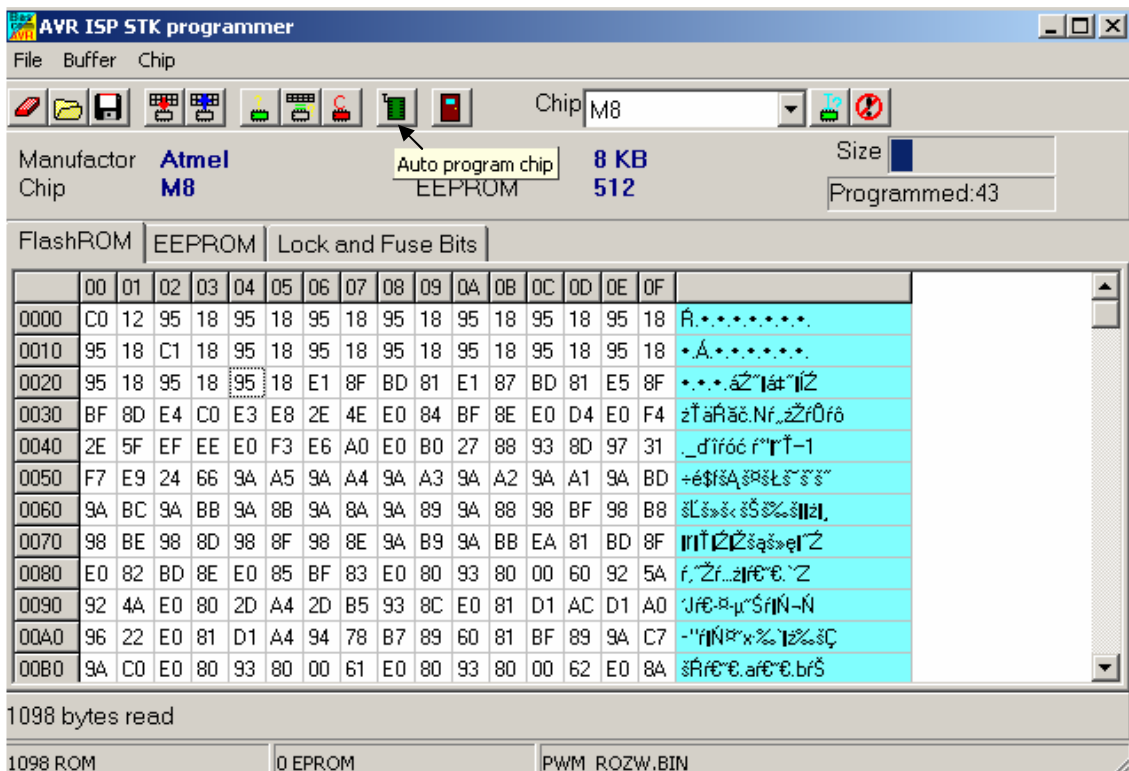
Program -> Compile (F7)

2. Ładowanie programu od kontrolera:

Aby dokonać wgrania programu do pamięci FLASH kontrolera należy z menu Program wybrać opcję Send to chip (F4).

Program -> Send to chip (F4)

Następnie w oknie programatora należy kliknąć ikonę *Auto program chip* (jak na rysunku poniżej).



Dodatek III

CALL Ustaw_pwm(wypełnienie)

Wywołanie procedury ustawiającej współczynnik wypełnienia modulatora PWM.

Gdzie:

Wypełnienie – zmienna lub stała 8 bitowa, wartość wypełnienia (0-255)

Opis:

Ustawia, do jakiej wartości będzie zliczał 8 bitowy timer1 z wartością 1 na wyjściu OC1A (PWM), potem do osiągnięcia maksymalnej wartości zliczania OC1A przejdzie w 0. Przy zmianie, z wartości maksymalnej (MAX_COUNT =255) na 0, ustawiona zostanie ponownie wartość 1, aż do osiągnięcia wartości W. Współczynnik wypełnienia można obliczyć jako W/MAX_COUNT .

DECR *zmienna*

Zmniejsza zawartość zmiennej o jeden.

gdzie:

zmienna - zmienna typu Byte, Integer, Word, Long lub Single

Opis:

Zawartość zmiennej jest zmniejszana o jeden. Instrukcja jest szybsza niż taka sama operacja z wykorzystaniem operatora -.

DIM *zmiana_1* As Byte

Deklaracja zmiennej 8 bitowej (liczby 8 bitowej)

DIM *zmiana_2* As Bit

Deklaracja zmiennej 1 bitowej

IF *wyrażenie* THEN

ciąg_instrukcji

END IF

IF *wyrażenie* THEN

ciąg_instrukcji

ELSE

ciąg_instrukcji

END IF

gdzie:

wyrażenie - testowane wyrażenie logiczne,

ciąg_instrukcji - dowolny ciąg instrukcji, wykonywanych po spełnieniu lub nie warunku określonego w wyrażeniu.

Opis:

Tworzy tzw. blok decyzyjny.

Instrukcja IF..THEN oblicza logiczną wartość podanego wyrażenia. Jeśli będzie ono prawdziwe (wynikiem będzie logiczna prawda) wykonany zostanie blok instrukcji umieszczony po instrukcji THEN. Jeśli będzie ono fałszywe, to instrukcje po słowie THEN nie zostaną wykonane. Wykonane za to będą instrukcje po słowie ELSE, jeśli ono występuje.

INCR *zmienna*

Zwiększa zawartość zmiennej o jeden.

gdzie:

zmienna - zmienna typu Byte, Integer, Word, Long lub Single

Opis:

Zawartość zmiennej jest zwiększana o jeden. Instrukcja jest szybsza niż taka sama operacja z wykorzystaniem operatora +.

Set Bit1

Ustawia określony bit na 1

Przykład:

Ustaw zmienną Bi1 na 1

DIM Bit1 as Bit

SET Bit1

Reset Bit1

Ustawia określony bit na 0

Przykład:

Ustaw zmienną Bi1 na 0

DIM Bit1 as Bit

RESET Bit1

WAITMS liczbaMS

Czekaj przez czas *liczbaMS* określony w ms

WHILE warunek

ciąg_instrukcji

WEND

Wykonuje określony ciąg instrukcji dopóki warunek jest spełniony.

gdzie:

warunek	wyrażenie obliczane jako warunek pętli,
ciąg_instrukcji	dowolny ciąg instrukcji języka BASCOM BASIC.

Opis:

Instrukcja służy do konstruowania pętli programowych. Gdy podany warunek jest prawdziwy (zwraca wartość logicznej prawdy) wtedy wykonywany będzie ciąg instrukcji umieszczony pomiędzy WHILE a WEND.

Wyjście z pętli jest możliwe tylko, gdy warunek nie będzie spełniony (zwróci wartość logicznego fałszu), lub też za pomocą instrukcji EXIT WHILE

Przykład:

Oczekiwanie na zmianę stanu klawisza z 1 na 0.

While Klawisz_lewo = 1

Wend

Operatory :

> większości ,
< mniejszości,
= równości

Tablica odniesień

zmienna = LOOKUP(*nr_elementu* , *etykieta*)

Zwraca wartość wybranego elementu w liniach DATA.

gdzie:

<i>zmienna</i>	<i>zmienna</i> , której przypisana będzie wartość podanego elementu,
----------------	--

nr_elementu numer elementu,
etykieta etykieta określająca adres danych umieszczonych w liniach DATA.

Opis:

Elementy są numerowane od zera. Numer elementu nie może być większy niż 65535.

Przykład :

Podstawienie za **W** drugiego elementu tablicy o nazwie Tablica_odniesień

$I = 2$

$W = \text{Lookup}(I, \text{Tablica_odniesień})$

Tablica_odniesień:

Data 0, 1, 2, 3

Final

XXX Ogólnopolska Olimpiada Wiedzy Elektrycznej i Elektronicznej
3 – 4 kwietnia 2006
Zespół Szkół Łączności w Krakowie

Zestaw pytań finałowych dla grupy elektronicznej

I	1. Działanie pamięci DRAM
	2. Porównanie własności szeregowego i równoległego stabilizatora napięcia
	3. Metody syntezy częstotliwości
II	1. Porównanie pamięci E ² PROM i flash
	2. Rodzaje szumów w układach elektronicznych
	3. Pomiar wartości skutecznej
III	1. Budowa i własności liczników zliczających w kodzie Johnsona
	2. Automatyczna regulacja wzmocnienia
	3. Port testowy JTAG w układach cyfrowych
IV	1. Hazardy w układach cyfrowych i sposoby ich eliminacji
	2. Przykłady realizacji filtrów pasmowo - przepustowych (układy bierne i czynne)
	3. Pomiar współczynnika zniekształceń nieliniowych
V	1. Działanie pamięci podręcznej w mikroprocesorze
	2. Porównanie modulacji amplitudy i częstotliwości
	3. Generatory „trójpunktowe” LC

VI	1. Synteza funkcji logicznych przy pomocy typowych bloków funkcjonalnych
	2. Porównaj filtry dolnoprzepustowe wg aproksymacji Butterwortha, Czebyszewa, Bessela i Cauera
	3. Układy zegara czuwania (watchdog) w mikrokontrolerach
VII	1. Kodowanie informacji na nośniku magnetycznym (dyskietka, dysk twardy)
	2. Generatory sterowane napięciem – przykłady realizacji
	3. Stabilność układów regulacji automatycznej
VIII	1. Układy logiki programowanej
	2. Zakłócenia od pól elektrycznych i magnetycznych – porównanie , środki zaradcze
	3. Układy zamkniętej pętli fazowej – działanie, zastosowanie
IX	1. Wyjaśnij, dlaczego zwiększenie liczby interfejsów wewnętrznych w mikrokontrolerze nie wymaga dodania nowych instrukcji
	2. Układy próbkująco – pamiętające
	3. Standard I ² C
X	1. Metody przetwarzania a/c
	2. Porównaj mikroprocesory typu RISC i CISC
	3. Standard One – wire
XI	1. Filtry eliminujące – realizacja, własności dynamiczne
	2. Standard przemysłowy HPIB (IEC 625, IEEE 488)
	3. Wykorzystaj porty szeregowo asynchroniczne do połączenia wielu mikrokontrolerów jednoukładowych