



„EUROELEKTRA”
Ogólnopolska Olimpiada Wiedzy Elektrycznej i Elektronicznej
Rok szkolny 2013/2014

Zadania z teleinformatyki na zawody II stopnia

Lp.	Zadanie
1.	<p>Na wejściu układu odbiornika SNR (stosunek sygnał szum) wynosi 20 dB. Ze względu na zbyt niski poziom sygnału wejściowego, na wejście wstawiono wzmacniacz o wzmacnieniu 20 dB i o współczynniku szumów $F = 2$.</p> <p>Jaką wartość przyjmie SNR1 na wejściu odbiornika po wstawieniu wzmacniacza?</p> <p>Układ przed zmianą: $\text{SNR} \rightarrow$ odbiornik</p> <p>Układ po zmianie: $\text{SNR} \rightarrow$ wzmacniacz $\xrightarrow{\text{SNR1}}$ odbiornik</p> <p><u>Rozwiązanie:</u> <u>Obliczenia:</u></p> $F = \frac{\text{SNR}}{\text{SNR1}}$ <p>Stąd $\text{SNR1} = \frac{\text{SNR}}{F}$</p> <p>Jako że 20 dB = 100 stąd $\text{SNR1} = \frac{100}{2} = 50$</p> <p>W mierze decybelowej $10\log 50 = 16,989$</p> <p><u>Odpowiedź:</u> $\text{SNR1} = 17 \text{ dB}$</p>
2.	<p>Dla pewnego toru symetrycznego obciążonego impedancją $R_1 = 120 \Omega$ współczynnik niedopasowania wynosi $q_1 = 0,1$.</p> <p>Jaką wartość musiałaby osiągnąć rezystancja obciążenia R_2 aby współczynnik niedopasowania osiągnął wartość $q_2 = 0,01$?</p>

	<p><u>Rozwiązanie:</u> <u>Obliczenia:</u> Z – impedancja toru R – impedancja obciążenia R_1 - rezystancja obciążenia dla q_1 R_2 - rezystancja obciążenia dla q_2</p> <p>$q = \frac{Z-R}{Z+R}$ gdzie: q- współczynnik niedopasowania</p> $q_1 = \frac{Z-R_1}{Z+R_1} \qquad q_2 = \frac{Z-R_2}{Z+R_2}$ <p>Z zależności q_1 wyznaczamy Z:</p> $Z = R_1 \left(\frac{1-q_1}{q_1-1} \right)$ $Z = 146,67 \, \Omega$ <p>Z zależności q_2 wyznaczamy R_2:</p> $R_2 = Z \left(\frac{1-q_2}{q_2+1} \right)$ $R_2 = 143,77 \, \Omega$ <p><u>Odpowiedź:</u> $R_2 = 143,77 \, \Omega$</p>										
3.	<p>Dla sygnału w formacie OFDM (<i>Orthogonal Frequency Division Multiplexing</i>) przyjęto odstęp ochronny 500 ns, zaś czas przetwarzania FFT (<i>Fast Fourier Transform</i>) zasadniczej części symbolu OFDM wynosi 2,5 μs.</p> <p>Jaką bitową przepływność uzyska się w paśmie 100 MHz przy zastosowaniu modulacji 256-QAM oraz kodu kanałowego o sprawności $\frac{1}{2}$? Odpowiedź powinna być poparta obliczeniami.</p> <p><u>Rozwiązanie:</u> <u>Dane:</u></p> <table> <tr> <td>Odstęp ochronny</td><td>$T_G = 500 \, \text{ns}$</td></tr> <tr> <td>Czas przetwarzania FFT</td><td>$T_{FFT} = 2,5 \, \mu\text{s}$</td></tr> <tr> <td>Szerokość kanału częstotliwościowego</td><td>$\Delta f = 100 \, \text{MHz}$</td></tr> <tr> <td>Wartościowość modulacji</td><td>$N = 256$</td></tr> <tr> <td>Sprawność kodu kanałowego</td><td>$K = \frac{1}{2}$</td></tr> </table> <p><u>Obliczenia:</u> Częstotliwościowy odstęp między podnośnymi OFDM:</p> $\Delta f_{FFT} = \frac{1}{T_{FFT}} = 400 \, \text{kHz}$ <p>Liczba podnośnych OFDM mieszcząca się w częstotliwościowym kanale Δf :</p> $L_{FFT} = \frac{\Delta f}{\Delta f_{FFT}} = 250$	Odstęp ochronny	$T_G = 500 \, \text{ns}$	Czas przetwarzania FFT	$T_{FFT} = 2,5 \, \mu\text{s}$	Szerokość kanału częstotliwościowego	$\Delta f = 100 \, \text{MHz}$	Wartościowość modulacji	$N = 256$	Sprawność kodu kanałowego	$K = \frac{1}{2}$
Odstęp ochronny	$T_G = 500 \, \text{ns}$										
Czas przetwarzania FFT	$T_{FFT} = 2,5 \, \mu\text{s}$										
Szerokość kanału częstotliwościowego	$\Delta f = 100 \, \text{MHz}$										
Wartościowość modulacji	$N = 256$										
Sprawność kodu kanałowego	$K = \frac{1}{2}$										

	<p>Zatem w miejscu modulatora zostanie zastosowana 256-punktowa transformata IFFT z 6 wygaszonymi podnośnymi.</p> <p>Łączny czas trwania symbolu OFDM:</p> $T_{OFDM} = T_G + T_{FFT} = 3\mu s$ <p>Liczba bitów przypadająca na jeden symbol modulacji pasmowej 256-QAM:</p> $L_b = \log_2 N = \log_2 256 = 8$ <p>Uzyskana łączna przepływność kanału OFDM:</p> $R_{OFDM} = \frac{L_b \cdot K \cdot L_{FFT}}{T_{OFDM}} \cong 333 \text{ Mb/s}$ <p>Odpowiedź: W paśmie przedstawionego w zadaniu kanału częstotliwościowego można uzyskać maksymalną przepływność bitową brutto wynoszącą około 333 Mb/s.</p>														
4.	<p>Moc EIRP (<i>Equivalent Isotropic Radiated Power</i>) ustalono systemowo na poziomie 200 mW. Poziom mocy na koncentrycznym wyjściu (radiowego nadajnika) karty bezprzewodowej wynosi 25 mW, zaś kierunkowy zysk anteny panelowej 20 dBi. Karta bezprzewodowa została połączona z anteną 10 metrowym kablem koncentrycznym o tłumienności, dla systemowej częstotliwości pracy, wynoszącej 45dB/100m.</p> <p>Jakie powinno być minimalne tłumienie dołączonego tłumika przy założeniu, że każde fizyczne połączenie dodatkowo wnosi tłumienie o wartości około 0,5 dB ?</p> <p>Rozwiązanie: Dane:</p> <table border="0"> <tr> <td>Moc EIRP:</td> <td>$P_{EIRP} = 200\text{mW}$</td> </tr> <tr> <td>Poziom mocy wyjściowej:</td> <td>$P_{wy} = 25\text{mW}$</td> </tr> <tr> <td>Kierunkowy zysk anteny panelowej:</td> <td>$G = 20\text{dBi}$</td> </tr> <tr> <td>Długość koncentrycznego kabla:</td> <td>$L = 10\text{m}$</td> </tr> <tr> <td>Tłumienność koncentrycznego kabla:</td> <td>$A_L = 45\text{dB}/100\text{m}$</td> </tr> <tr> <td>Tłumienie złączki:</td> <td>$A_z = 0,5\text{dB}$</td> </tr> <tr> <td>Liczba zastosowanych złązek:</td> <td>$n = 3$</td> </tr> </table> <p>Obliczenia: Poziom EIRP przeliczamy na dBm:</p> $P_{EIRP} = 10 \log \frac{200\text{mW}}{1\text{mW}} = 23,01 \text{ dBm}$ <p>Poziom mocy wyjściowej przeliczamy na dBm:</p> $P_{wy} = 10 \log \frac{25\text{mW}}{1\text{mW}} = 13,98 \text{ dBm}$ <p>Liczymy bilans tłumienia transmisyjnego łączy układu antenowego:</p> $P_{EIRP} = P_{wy} + G - A \Rightarrow A = P_{wy} + G - P_{EIRP}$ $A = 13,98 \text{ dBm} + 20 \text{ dBi} - 23,01 \text{ dBm} = 10,97 \text{ dB}$ <p>Liczymy tłumienie kabla koncentrycznego o tłumienności w przeliczeniu na metr bieżący $A_{Lm} = 0,45 \text{ dB/m}$ wraz ze złączkami:</p> $A_{kz} = A_{Lm} \cdot L + n \cdot A_z = 0,45 \text{ dB/m} \cdot 10 \text{ m} + 3 \cdot 0,5\text{dB} = 6 \text{ dB}$ <p>Liczymy minimalne tłumienie tłumika mikrofalowego:</p> $A_r = A - A_{kz} = 10,97 \text{ dB} - 6 \text{ dB} = 4,97 \text{ dB} \cong 5 \text{ dB}$	Moc EIRP:	$P_{EIRP} = 200\text{mW}$	Poziom mocy wyjściowej:	$P_{wy} = 25\text{mW}$	Kierunkowy zysk anteny panelowej:	$G = 20\text{dBi}$	Długość koncentrycznego kabla:	$L = 10\text{m}$	Tłumienność koncentrycznego kabla:	$A_L = 45\text{dB}/100\text{m}$	Tłumienie złączki:	$A_z = 0,5\text{dB}$	Liczba zastosowanych złązek:	$n = 3$
Moc EIRP:	$P_{EIRP} = 200\text{mW}$														
Poziom mocy wyjściowej:	$P_{wy} = 25\text{mW}$														
Kierunkowy zysk anteny panelowej:	$G = 20\text{dBi}$														
Długość koncentrycznego kabla:	$L = 10\text{m}$														
Tłumienność koncentrycznego kabla:	$A_L = 45\text{dB}/100\text{m}$														
Tłumienie złączki:	$A_z = 0,5\text{dB}$														
Liczba zastosowanych złązek:	$n = 3$														

	<p><u>Odpowiedź:</u></p> <p>W celu zachowania poziomu EIRP, minimalne tłumienie tłumika znajdującego się w mikrofalowej linii transmisyjnej powinno wynosić około 5 dB.</p>																																	
5.	<p>Proszę napisać funkcję podającą liczbę całkowitą wylosowaną spośród liczb od 0 do n włącznie, gdzie n jest argumentem funkcji. W drugim kroku proszę napisać program main, który będzie wykorzystywał napisaną funkcję do zbadania rozkładu liczb wylosowanych z przedziału [0, 10]. Liczba prób losowania ma być ustalana z wejścia, a zadaniem programu jest wypisanie informacji, ile razy wystąpiły poszczególne liczby.</p> <p>Proszę wykorzystać funkcję rand, która w języku C zwraca pseudolosową liczbę całkowitą z zakresu [0, RAND_MAX]. Stałą RAND_MAX powinna być zdefiniowana w stdlib.h, a jej wartość nie powinna być mniejszą niż 32767 (górna granica dla typu int).</p> <p>Przykład:</p> <p>Podaj liczbę prób losowania: 1000</p> <p>Wynik:</p> <table><tr><td>liczbę</td><td>0</td><td>wylosowano 80 razy</td></tr><tr><td>liczbę</td><td>1</td><td>wylosowano 89 razy</td></tr><tr><td>liczbę</td><td>2</td><td>wylosowano 105 razy</td></tr><tr><td>liczbę</td><td>3</td><td>wylosowano 95 razy</td></tr><tr><td>liczbę</td><td>4</td><td>wylosowano 106 razy</td></tr><tr><td>liczbę</td><td>5</td><td>wylosowano 84 razy</td></tr><tr><td>liczbę</td><td>6</td><td>wylosowano 90 razy</td></tr><tr><td>liczbę</td><td>7</td><td>wylosowano 78 razy</td></tr><tr><td>liczbę</td><td>8</td><td>wylosowano 81 razy</td></tr><tr><td>liczbę</td><td>9</td><td>wylosowano 93 razy</td></tr><tr><td>liczbę</td><td>10</td><td>wylosowano 99 razy</td></tr></table> <p><u>Rozwiązanie:</u></p> <pre>#include <stdio.h> #include <stdlib.h> #define N 10 main() { int przyp(int); int lb_los, t[N+1], i; printf("Podaj liczbę prób losowania: "); scanf("%d", &lb_los); for(i=0; i<=N; i++) { t[i] = 0; } for(i=1; i<=lb_los; i++) { t[przyp(N)]++; } for(i=0; i<=N; i++) { printf("liczbe %4d wylosowano %6d razy\n", i, t[i]); } }</pre>	liczbę	0	wylosowano 80 razy	liczbę	1	wylosowano 89 razy	liczbę	2	wylosowano 105 razy	liczbę	3	wylosowano 95 razy	liczbę	4	wylosowano 106 razy	liczbę	5	wylosowano 84 razy	liczbę	6	wylosowano 90 razy	liczbę	7	wylosowano 78 razy	liczbę	8	wylosowano 81 razy	liczbę	9	wylosowano 93 razy	liczbę	10	wylosowano 99 razy
liczbę	0	wylosowano 80 razy																																
liczbę	1	wylosowano 89 razy																																
liczbę	2	wylosowano 105 razy																																
liczbę	3	wylosowano 95 razy																																
liczbę	4	wylosowano 106 razy																																
liczbę	5	wylosowano 84 razy																																
liczbę	6	wylosowano 90 razy																																
liczbę	7	wylosowano 78 razy																																
liczbę	8	wylosowano 81 razy																																
liczbę	9	wylosowano 93 razy																																
liczbę	10	wylosowano 99 razy																																

	<pre> } } int lprzyp (int n) { int i; i = rand() / (RAND_MAX + 1.)*(n+1); return (i); } </pre>
6.	<p>Założmy, że mamy do czynienia z następującym programem:</p> <pre> #include <stdio.h> main() { int i, n, sum; sum = 0; for (i=0; i<4; i++) { printf("Podaj liczbę całkowitą "); scanf("%d", &n); sum += n; } printf("Suma %d\n", sum); } </pre> <p>Proszę napisać program, który będzie realizował tę samą funkcjonalność, jednak zamiast instrukcji for będzie wykorzystywał instrukcję do ... while.</p> <p><u>Rozwiązanie:</u></p> <pre> #include <stdio.h> main() { int i, n, sum; sum = 0; i = 0; do { printf("Podaj liczbę całkowitą "); scanf("%d", &n); sum += n; i++; } while (i < 4); printf("Suma %d\n", sum); } </pre>

<p><i>Opracowali:</i> dr inż. Zbigniew Zakrzewski dr inż. Mariusz Aleksiewicz mgr inż. Jan Kołodziej</p>	<p><i>Sprawdził:</i> dr inż. Jacek Majewski</p>	<p><i>Zatwierdził:</i> Przewodniczący Rady Naukowej Olimpiady dr inż. Sławomir Cieślik</p>
---	--	---