

YILDIZ TEKNİK ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

AMPLİFİKATÖRLERİN BİLGİSAYARLA ANALİZ VE TASARIMI

Elek.Müh. Mehmet YUMURTACI

F.B.E. Elektrik Mühendisliği Anabilim Dalında
hazırlanan

YÜKSEK LİSANS TEZİ

Tez Danışmanı : Prof.Dr. Halit PASTACI

T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMENTASYON MERKEZİ

İSTANBUL, 1993

İÇİNDEKİLER

	<u>Sayfa No</u>
İÇİNDEKİLER	iii
TEŞEKKÜR	iv
ÖZET	v
ABSTRACT	vi
1. GİRİŞ	1
2. TEK KATLI AMPLİFİKATÖRLERİN ORTA FREKANSLARDAKİ ANALİZİ	3
2.1. Bipolar Transistörlü Amplifikatörün Analizi	3
2.2. JFET veya Artırılmış Mosfetli Amplifikatörün Analizi	9
2.3. Artırılmış Mosfetli Amplifikatörün Analizi	15
3. TEK KATLI AMPLİFİKATÖRÜN ALÇAK FREKANSLARDAKİ ANALİZİ	17
3.1. Bipolar Transistörlü Amplifikatörün Alçak Frekans Cevabı	17
3.2. FET li Amplifikatörün Alçak Frekans Cevabı	21
4. TEK KATLI AMPLİFİKATÖRÜN YÜKSEK FREKANSLARDAKİ ANALİZİ	22
4.1. Bipolar Transistörlü Amplifikatörün Yüksek Frekans Cevabı	22
4.2. FET li Amplifikatörün Yüksek Frekans Cevabı	23
5. TEK KATLI AMPLİFİKATÖRÜN TASARIMI	27
5.1. Bipolar Transistörlü Amplifikatörün Tasarımı	27
5.2. FET li Amplifikatörün Tasarımı	33
6. TEK KATLI BİR AMPLİFİKATÖRÜN ANALİZ VE TASARIMINI YAPAN BİLGİSAYAR PROGRAMI	36
7. BİLGİSAYARDAN ALINAN SONUÇLAR	65
8. KAYNAKLAR	89
9. ÖZGEÇMİŞ	90

TEŞEKKÜR

Bu tezin hazırlanması sırasında önerilerinden istifade ettiğim sayın hocam Prof. Dr. Halit Pastacı ' ya teşekkür ederim.



ÖZET

Bu tezin hazırlanmasında tek katlı amplifikatörler üzerinde durulmuştur. Bipolar transistör, jfet, mosfet ve artırılmış mosfet elemanları kullanılarak oluşturulabilecek amplifikatörlerin analiz ve tasarımlarının rahatlıkla yapılması amaçlanmıştır. Yukarıdaki dört elemandan herbirisi için ortak emetör, ortak baz ve ortak kollektörü bağlantılar ayrı ayrı ele alınmıştır.

Amplifikatörlerin analizi sırasında transistörlerin dc eşdeğer devresi ve hibrit π modeli kullanılmıştır. Herbir model için dc analiz yapıldıktan sonra kazanç ve empedans ifadeleri çıkarılmıştır. Pascal dili ile yazılan bilgisayar programında söz konusu her bir bağlantı için analiz ve tasarım hesapları yapılmıştır. Program tüm seçenekleri içeren bir menü ile desteklenerek kullanıcıya kolaylık sağlanmıştır.

ABSTRACT

While this thesis has been prepared , the main subject is the amplifiers which consist of one transistor. The goal is to easily analyze and to design the amplifiers which can be composed of bipolar transistors, jfets, mosfets and enhanced mosfets. For each of the four elements above, the different connections with the common collector, the common gates and the common emitters are separately examined.

During the analyses of amplifiers, equivalent dc circuits of transistors and hybrid π models have been used . After dc analyses of each models advantage and impedance equations have been derived. The programming language used in this program is pascal. In this computer program, for each connections, design and analyses computations have been done. This program has been supported by menus of every choice. This makes easy to use this computer program.

1. GİRİŞ

1.1. Transistör

Transistörler akım kontrol elemanı olarak kullanılan yarı iletken elemanlardır. Transistörlerin pek çok çeşitleri olmakla beraber bipolar transistör ve alan etkili transistör olmak üzere iki ana gruba ayrılırlar. Bipolar transistörler npn ve pnp tipinde olabilir. npn tipi bir transistör birbirinden çok ince p tipi bir yarı iletken tabaka ile ayrılan iki n tipi yarı iletken oluşur. pnp tipi transistör ise iki p tipi yarı iletken arasına bir n tipi yarı iletken yerleştirilmesiyle oluşan bir düzendir. Her iki transistörde de aradaki yarı iletken tabakaya baz, diğer yarı iletken tabakalardan birisine emetör öbürüne kollektör denir.

Alan etkili transistörler ise bir yarı iletkenin içinden akan akımın bir elektriksel alan yardımı ile kontrol edilmesi prensibine dayanırlar. Kısaca FET diye anılan alan etkili transistörler iki sınıfa ayrılırlar : Jonksiyonlu alan etkili transistör (JFET) ve MOSFET ler. Alan etkili transistörler çok büyük giriş direncine sahip yarı iletken devre elemanlarıdır. Bipolar transistörlerle genellikle FET lere göre daha fazla kazanç sağlanabilir. Fakat bunların giriş dirençleri küçüktür. Ayrıca MOSFET ler yapımlarının kolaylığı ile beraber digital tümdevrelerde bir yarı iletken parçacığına daha çok sayıda eleman sığdırmaya elverişlidirler.

1.2. Amplifikatörler

Amplifikatör, bir işaret kaynağı tarafından girişine verilen güç, çıkışından alınan işaret gücüne göre daha küçük olan, yani güç kazancı sağlayan bir devredir.

Amplifikatörlerin temel özellikleri güç kazancı sağlamak olduğu halde V_2 nin V_1 den büyük olması halinde yahut bu özelliğin önemli olduğu hallerde devreye gerilim amplifikatörü yahut I_2 nin I_1 den daha büyük olmasının önemli olduğu hallerde devreye akım amplifikatörü denir. Fakat daima bir güç kazancı mevcuttur.

Herhangibir transistör için akım, gerilim ve güç kazancı ayrı ayrı hesaplanabilir. Bu kazançları sırasıyla A_i , A_v ve A_p ile gösterirsek

$$A_i = \frac{I_2}{I_1}, \quad A_v = \frac{V_2}{V_1}, \quad A_p = \frac{P_2}{P_1} \quad \text{şeklinde olur.}$$

Genel olarak A_i ve A_v frekansa bağlı büyüklüklerdir. V_1 in genliği sabit tutulup frekansı değiştirilirse V_2 nin genliği ve V_2 ile V_1 arasındaki faz farkı değişir. Akım veya gerilim kazancının pek fazla değişmediği frekans bandı o amplifikatörün normal kullanma bölgesidir. Bu frekanslar için A_v ve A_i yaklaşık olarak reeldir ve güç kazancı

$A_p = A_i A_v$ yazılabilir.

Amplifikatörlerde amaca göre çeşitli bağlantı şekilleri kullanılabilir. Transistörün baz ucu hem giriş hem de çıkışta ortak ise ortak bazlı bağlantı, emetör ucu giriş ve çıkışta ortak ise ortak emetörlü bağlantı denir. Ortak kollektörlü devrede ise yük direnci kollektör yerine emetöre bağlanır.

Ortak emetörlü devrede yüksek akım ve güç kazancı, iyi bir gerilim kazancı ile orta değerlerde giriş ve çıkış empedansı elde edilir. Ortak bazlı devrede orta değerli gerilim kazancı elde edilip giriş direnci küçüktür. Ortak kollektörlü devrede ise gerilim kazancı 1 den küçük olup çıkış empedansı küçüktür.



2. TEK KATLI AMPLİFİKATÖRLERİN ORTA FREKANSLARDAKİ ANALİZİ

Bu bölümde bir transistörlü tek katlı amplifikatörün analizi incelenecektir. Bipolar transistör, fet ve mosfet kullanılarak oluşturulabilecek olan devrelerin alçak frekanslarda, orta frekanslarda ve yüksek frekanslardaki gerilim kazancı, akım kazancı, güç kazancı, giriş ve çıkış empedansları ayrı ayrı hesaplanacaktır.

2.1. Bipolar Transistörlü Amplifikatörün Analizi

Tek katlı ortak emetörlü amplifikatörün kazanç ve empedans hesabına geçmeden dc çalışma noktasının bulunması gerekir. Bulunacak bu ifadeler ortak baz ve ortak kollektörlü ifadeler için de geçerlidir. Ortak emetörlü devrenin şekil 2.1.b de gösterilen dc kutuplama devresi ve yaklaşık eşdeğer devresine bakarak aşağıdaki eşitlikleri yazabiliriz.

$$V_{th} = R_{th} + I_B + R_{BB}I_B + V_o + R_E(I_B + I_C)$$

$$I_C = \beta_{dc}I_B + I_{CEO} + \frac{V_{CE}}{R_o}$$

$$I_B \frac{1}{\beta_{dc}} I_C - \frac{1}{\beta_{dc}} I_{CEO} - \frac{1}{\beta_{dc} R_o} V_{CE}$$

$$V_{th} = (R_{th} + R_{BB} + R_E)I_B + R_E I_C + V_o$$

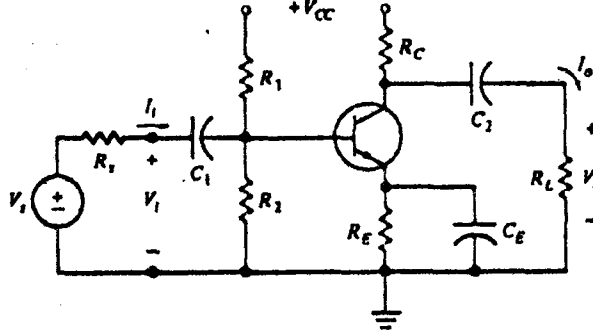
$$\frac{R_{th}+R_{BB}+R_E}{\beta_{dc}R_o} V_{CE} - \left(\frac{R_{th}+R_{BB}+R_E}{\beta_{dc}} + R_E \right) I_C = V_o - \frac{R_{th}+R_{BB}+R_E}{\beta_{dc}} I_{CEO} - V_{th}$$

$$\left(1 - \frac{R_E}{\beta_{dc}R_o} \right) V_{CE} + \left(R_C + R_E + \frac{R_E}{\beta_{dc}} \right) I_C = V_{CC} + \frac{R_E}{\beta_{dc}} I_{CEO}$$

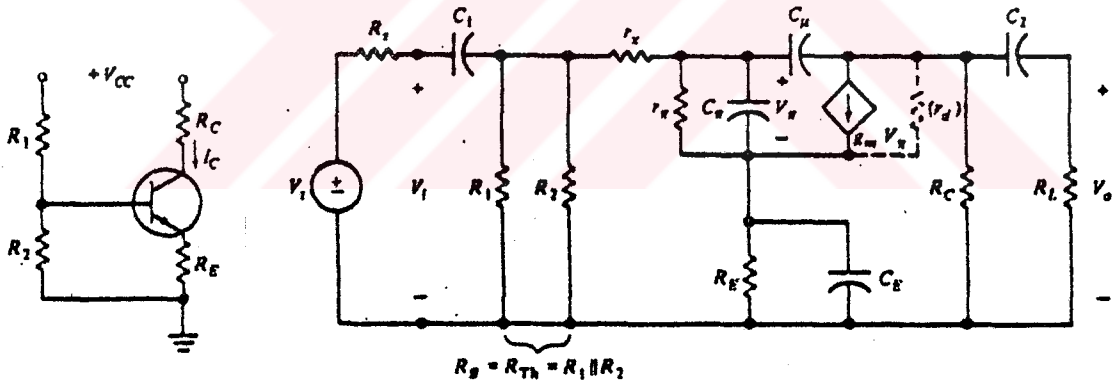
Yukarıdaki iki eşitlikten V_{CE} ve I_C çözülür. Hibrit π modelindeki parametrelerin hesabı için:

Bipolar transistörün $V_{BE} - I_B$ eğisi diyot eğrisine benzemektedir.

$$I_B = I_{BO}(e^{qV_{BE}/nkT} - 1) = I_{BO}e^{qV_{BE}/nkT} \quad \text{yazılabilir.}$$



(a)



(b)

Şekil 2.1

- Tek katlı ortak emetörlü amplifikatör devresi.
- Ortak emetörlü amplifikatörün dc kutuplama ve eşdeğer devresi.

$$g_m = \frac{i_c}{v_{be}} \Big|_{V_{CE}=0}$$

diyottaki iletkenliğe benzeyen bu ifade i_c ile v_{be} arasındaki dinamik iletkenliği göstermektedir.

$$I_B = \frac{1}{\beta_{dc}} I_C - \frac{1}{\beta_{dc}} I_{CEO} - \frac{1}{\beta_{dc}} \frac{V_{CE}}{R_o} = I_{BO} e^{qV_{BE}/nkT}$$

$$I_C = \beta_{dc} I_{BO} e^{qV_{BE}/nkT} + I_{CEO} + \frac{V_{CE}}{R_o}$$

$$g_m = \frac{i_c}{v_{be}} \Big|_{V_{CE}=0}$$

$$I_C = \beta_{dc} I_{BO} e^{qV_{BE}/nkT} + I_{CEO}$$

$$g_m = \frac{i_c}{v_{be}} \Big|_Q = \frac{i_c}{v_{be}} = \frac{q}{nkT} \beta_{dc} I_{BO} e^{qV_{BE}/nkT}$$

$$g_m = \frac{q}{nkT} (I_C - I_{CEO})$$

I_C akımı npn transistöründe pozitif pnp de ise negatiftir. g_m ise daima pozitif değere sahiptir. Bundan dolayı

$$g_m = \frac{q}{nkT} |I_C - I_{CEO}| \quad n = 1 \text{ ve } T = 25^\circ\text{C için}$$

$$g_m = 38.92 |I_C - I_{CEO}| \quad \text{olur.}$$

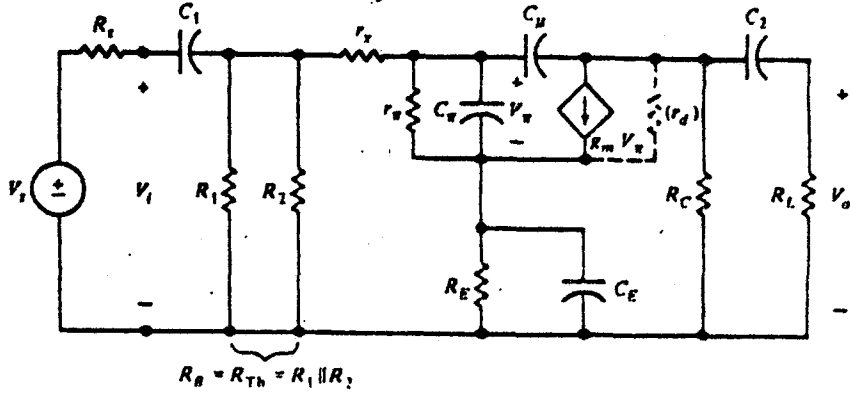
Ortak emetörlü bağlantı için şekil 2.2.b deki orta frekans eşdeğer devresinden

$$A_{vi} = \frac{V_o}{V_i}, \quad V_\pi = -V_i$$

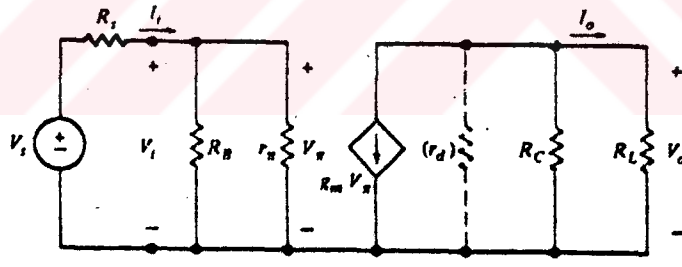
$$V_o = -g_m V_\pi (r_d // R_C // R_L) = -g_m V_i (r_d // R_C // R_L)$$

$$A_{vi} = -g_m(r_d // R_C // R_L)$$

$$R_{BB} = R_{th} = R_1 // R_2$$



(a)



(b)

Şekil 2.2

a. Ortak emetörlü amplifikatörün küçük işaret eşdeğer devresi.

b. Orta frekans eşdeğeri.

$$V_i = V_s \frac{(r_{\pi} // R_B)}{R_s + (r_{\pi} // R_B)}$$

$$A_{vs} = \frac{V_o}{V_s} = \frac{V_o}{V_i} \frac{V_i}{V_s} = A_{vi} \frac{V_o}{V_s}$$

$$A_{vs} = A_{vi} \frac{r_{\pi}/R_B}{R_s + (r_{\pi}/R_B)}$$

$$A_{vs} = -g_m(r_d//R_C//R_L) \frac{r_{\pi}/R_B}{R_s + (r_{\pi}/R_B)}$$

$$Z_i = \frac{V_i}{I_i}$$

$$Z_i = r_{\pi}/R_B$$

$$A_I = \frac{I_o}{I_i} = \frac{V_o/R_L}{V_i/Z_i} = A_{vi} \frac{Z_i}{R_L}$$

$$A_p = \frac{V_o I_o}{V_i I_i} = A_{vi} A_i = (A_{vi})^2 \frac{Z_i}{R_L}$$

$$Z_o = (R_C//r_d)$$

Şekil 2.3.a da gösterilen ortak bazlı devre için ise, orta frekans eşdeğer devresinden

$$V_{\pi} = V_i$$

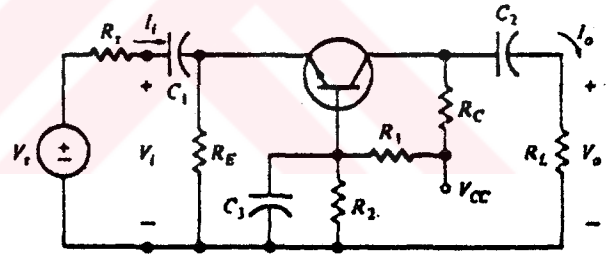
$$V_o = -V_{\pi} \left[g_m + \frac{1 - g_m(R_C//R_L)}{r_d + (R_C//R_L)} \right] (R_C//R_L)$$

$$Z_i = \frac{V_i}{I_i} = R_E//r_{\pi}//z_r$$

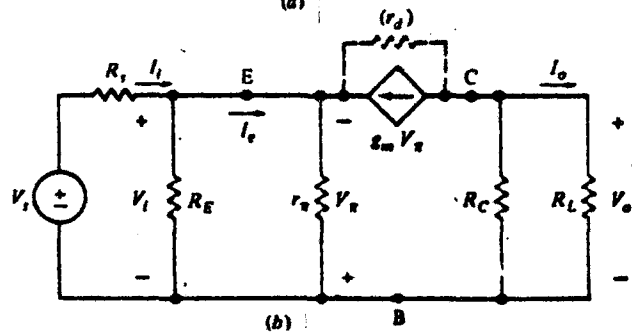
$$z_r = \frac{1}{g_m} \frac{r_d(R_C//R_L)}{r_d + 2(R_C//R_L)}$$

$$Z_i = R_E//r_{\pi} // \frac{1}{g_m} \frac{r_d + (R_C//R_L)}{r_d + 2(R_C//R_L)}$$

$$A_{vs} = \frac{V_o}{V_s} = A_{vi} \frac{Z_i}{R_s + Z_i}$$



(a)

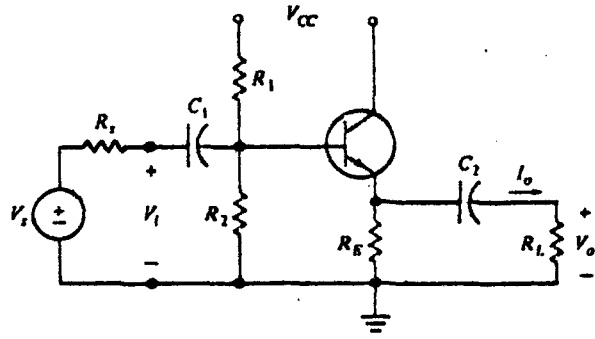


(b)

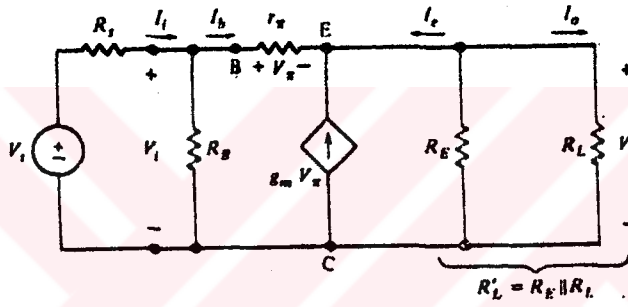
Şekil 2.3

a. Tek katlı ortak bazlı amplifikatör devresi

b. Orta frekans eşdeğeri.



(a)



(b)

Şekil 2.4

a. Tek katlı ortak kollektörlü amplifikatör devresi.

b. Orta frekans eşdeğeri.

$$A_I = \frac{I_o}{I_i} = \frac{V_o/R_L}{V_i/Z_i} = A_{vi} \frac{Z_i}{R_L}$$

$$A_p = \frac{V_o I_o}{V_i I_i} = (A_{vi})^2 \frac{Z_i}{R_L}$$

$$Z_o = \frac{V_{oc}}{I_{sc}} = R_c - \frac{1 - g_m(R_c // R_L)}{g_m[r_d + (R_c // R_L)]} R_c$$

Ortak kollektör bağlantılı devrede şekil 2.4.b deki eşdeğer devre yardımıyla kazanç ve empedans hesabı aşağıda yapılmıştır.

$$R_L' = R_E // R_L$$

$$Z_i = R_B // (r_\pi + z_r)$$

$$V_\pi = r_\pi I_b, \quad g_m V_\pi = g_m r_\pi I_b = \beta_o I_b$$

$$V_o = (I_b + \beta_o I_b) R_L'$$

$$z_r = \frac{V_o}{I_b} = (\beta_o + 1) R_L'$$

$$Z_i = \frac{V_i}{I_i} = R_B // [r_\pi + (1 + \beta_o) R_L']$$

$$V_o = V_i \frac{z_r}{r_\pi + z_r}$$

$$A_{vi} = \frac{V_o}{V_i} = \frac{z_r}{r_\pi + z_r} = \frac{(1 + \beta_o) R_L'}{r_\pi + (1 + \beta_o) R_L'}$$

$$A_i = \frac{I_o}{I_i} = \frac{V_o / R_L}{V_i / Z_i} = A_{vi} \frac{Z_i}{R_L}$$

$$A_p = \frac{V_o I_o}{V_i I_i} = A_{vi} A_i = (A_{vi})^2 \frac{Z_i}{R_L}$$

$$Z_o = r_d // R_E$$

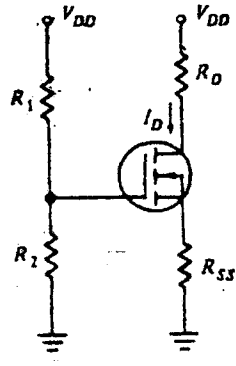
2.2. Jfet Veya Arıtılmış Mosfetli Amplifikatörün Analizi

Fetli amplifikatörde de kazanç ve empedans hesaplarından önce dc devrenin dc analizinin yapılması gerekir. Bu amaç için şekil 2.5 i kullanabiliriz. n kanallı jfet için;

$$I_D = D \left[(V_{GS} - V_p) V_{DS} - \frac{V_{DS}^2}{2} \right]$$

$$I_{DSS} = I_D \quad (V_{DS} = -V_p, \quad V_{GS} = 0 \quad \text{ için })$$

Doyma bölgesinde çalışma için $V_{DS} \geq V_{GS} - V_p$ olması gerekir.



Şekil 2.5 Ortak emetörlü ve JFET içeren amplifikatörün dc kutuplama devresi.

$V_{DS} \geq V_{GS} - V_p$ için sınır değer olarak $V_{DS} = V_{GS} - V_p$ alınabilir.

$$I_D = D \left[(V_{GS} - V_p)(V_{GS} - V_p) - \frac{(V_{GS} - V_p)^2}{2} \right]$$

$$I_D = \frac{D}{2}(V_{GS} - V_p)^2, \quad (V_{DS} \geq V_{GS} - V_p \quad \text{ve} \quad V_p \leq V_{GS} \leq 0 \quad \text{için})$$

$$I_D|_{V_{GS}=0} = I_{DSS} = \frac{D}{2}(-V_p)^2, \quad D = \frac{I_{DSS}}{V_p^2}$$

$$I_D = \frac{I_{DSS}}{V_p^2}(V_{GS} - V_p)^2, \quad V_{DS} \geq V_{GS} - V_p \quad \text{ve} \quad V_p \leq V_{GS} \leq 0 \quad \text{için}$$

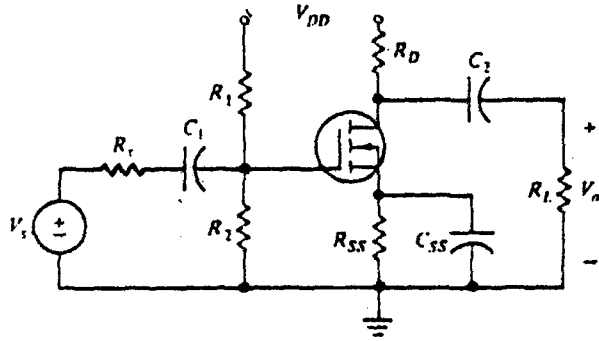
$$V_{DS} = V_{DD} - R_D I_D - R_{SS} I_D$$

$$I_D = \frac{I_{DSS}}{V_p^2}(V_{GS} - V_p)^2$$

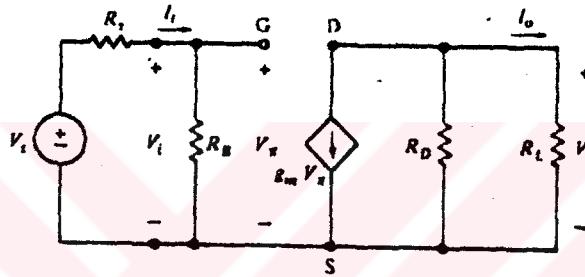
$$V_{GS}^2 - 2V_p V_{GS} + V_p^2 - \frac{I_D V_p}{I_{DSS}} = 0$$

$$\Delta = 4V_p^2 - 4V_p^2 \left(1 - \frac{I_D}{I_{DSS}}\right)$$

$$\Delta = 4V_p^2 \frac{I_D}{I_{DSS}}$$



(a)



(b)

Şekil 2.6

- a. Jfet içeren ortak emetörlü bir amplifikatör devresi.
b. Orta frekans eşdeğeri.

$$V_{GS} = \frac{2V_p \pm 2V_p \sqrt{\frac{I_D}{I_{DSS}}}}{2}$$

$V_{GS} \leq V_F$ olması gerektiğinden

$$V_{GS} = V_p - V_p \sqrt{\frac{I_D}{I_{DSS}}}$$

$$g_m = \left. \frac{di_D}{dV_{GS}} \right|_Q \quad \text{şeklinde yazılabilir}$$

$$I_D = \frac{I_{DSS}}{V_p^2} (V_{GS} - V_p)^2 \quad \text{idi. } I_D \text{ yerine } i_D, V_{GS} \text{ yerine } v_{GS} \text{ konularak}$$

$$i_D = \frac{I_{DSS}}{v_p^2} (v_{GS} - V_p)^2$$

$$g_m = \frac{2I_{DSS}}{V_p^2} (v_{GS} - V_p) \quad \text{elde edilir.}$$

Bulunan bu değerler ortak emetör, ortak baz ve ortak kollektörlü devreler için aynen kullanılacaktır.

Ortak emetörlü amplifikatörün kazanç ve empedans ifadeleri şekil 2.6.b deki eşdeğer devre yardımıyla hesaplanabilir.

$$V_o = -g_m V_i (R_D // R_L) = -g_m V_i (R_D // R_L)$$

$$A_{vi} = \frac{V_o}{V_i} = -g_m (R_D // R_L)$$

$$A_{vs} = \frac{V_o}{V_s} = \frac{V_o}{V_i} \frac{V_i}{V_s} = A_{vi} \frac{R_B}{R_s + R_B}$$

$$A_{vs} = -\frac{R_B}{R_s + R_B} g_m (R_D // R_L)$$

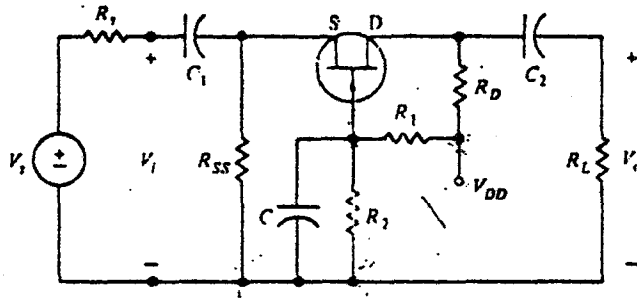
$$Z_i = \frac{V_i}{I_i} = R_B$$

$$A_i = A_{vi} \frac{R_B}{R_L}$$

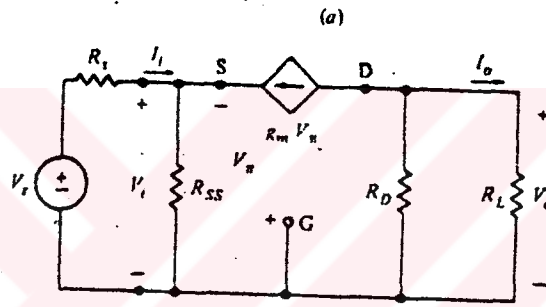
$$A_p = \frac{V_o I_o}{V_i I_i} = A_{vi} A_i = A_{vi}^2 \frac{R_B}{R_L}$$

$$Z_o = R_D$$

Şekil 2.7.a daki ortak bazlı amplifikatörün orta frekanslardaki bağıntıları da devrenin şekil 2.7.b deki orta frekans eşdeğerine bakarak aşağıdaki gibi çıkarılabilir.



(a)



(b)

Şekil 2.7

- a. Ortak bazlı fetli amplifikatör devresi.
b. Orta frekans eşdeğeri.

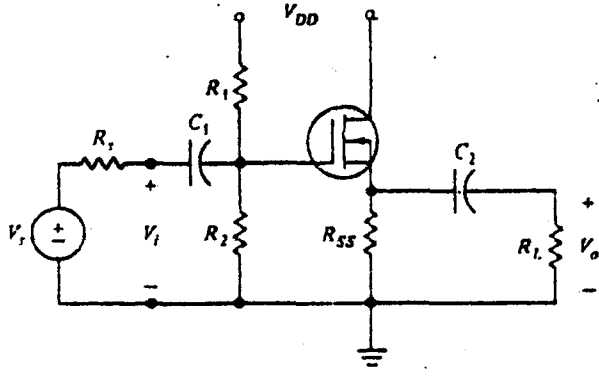
$$A_{vi} = g_m(R_D // R_L)$$

$$Z_i = R_{SS} // 1/g_m$$

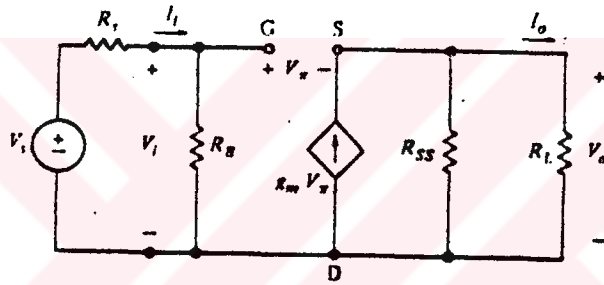
$$A_{vs} = A_{vi} \frac{Z_i}{R_s + Z_i}$$

$$A_I = A_{vi} \frac{Z_i}{R_L}$$

$$Z_o = R_D$$



(a)



(b)

Şekil 2.8

- a. Bir adet jfet kullanılarak gerçekleştirilmiş ortak kollektörlü amplifikatör.
b. Orta frekans eşdeğer devresi.

Ortak kollektörlü bağlantı için de şekil 2.8.b den benzer şekilde

$$R'_L = R_{SS} // R_L$$

$$V_o = g_m V_{gs} R'_L = g_m (V_i - V_o) R'_L$$

$$A_{vi} = \frac{V_o}{V_i} = \frac{g_m R'_L}{1 + g_m R'_L}$$

$$Z_i = \frac{V_i}{I_i} = R_B$$

$$A_{vs} = \frac{V_o}{V_s} = A_{vi} \frac{R_B}{R_s + R_B}$$

$$A_i = \frac{V_o/R_L}{V_i/Z_i} = A_{vi} \frac{R_B}{R_L}$$

$$A_p = A_{vi}^2 \frac{R_B}{R_L}$$

$$Z_o = R_{SS} // 1/g_m$$

2.3. Artırılmış Mosfetli Amplifikatörün Analizi

Artırılmış mosfet kullanılarak gerçekleştirilen amplitudin orta frekanslardaki analizi jfetli ampli ile tamamen aynı olduğundan dc analiz için gerekli olan bağımlıların çıkarılmasıyla yetinilecektir.

Artırılmış mosfette doyma ve omik bölgeler arasındaki sınır $V_{DS} = V_{GS} - V_T$ dir.

$$I_D = K \left[(V_{GS} - V_T) V_{DS} - \frac{V_{DS}^2}{2} \right]$$

$$I_D = K \left[(V_{GS} - V_T)(V_{GS} - V_T) - \frac{(V_{GS} - V_T)^2}{2} \right]$$

$$I_D = \frac{K}{2} (V_{GS} - V_T)^2, \quad V_{DS} \geq V_{GS} - V_T \quad \text{ve} \quad V_{GS} \geq V_T \quad \text{için.}$$

$$V_G = V_{DD} \frac{R_2}{R_1 + R_2}, \quad V_{GS} = V_G - R_{SS} I_D$$

$$V_{DS} = V_{DD} - (R_D + R_{SS}) I_D$$

Hibrid π parametreleri ise,

$$i_D = \frac{K}{2} (V_{GS} - V_T)^2, \quad V_{DS} \geq V_{GS} - V_T \quad \text{ve} \quad V_{GS} \geq V_T$$

$$g_m = \left. \frac{di_D}{dV_{GS}} \right|_Q = K(V_{GS} - V_T) \Big|_Q$$

$$g_m = K(V_{GS} - V_T)$$



3. TEK KATLI AMPLİFİKATÖRÜN ALÇAK FREKANSLARDAKİ ANALİZİ

Orta frekans analizinde kısa devre edilen kuplaj ve köprüleme kapasiteleri alçak frekanslarda etkili olduğundan böyle bir sadeleştirme yapılamaz. Amplinin alçak frekans analizinde hedef köşe frekanslarını bulmak olacaktır.

3.1. Bipolar Transistörlü Amplifikatörün Alçak Frekans Cevabı

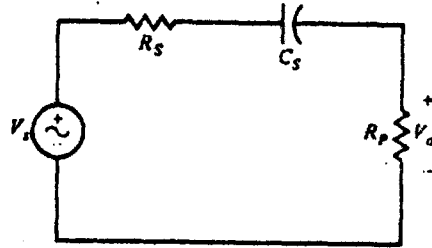
Amplifikatörü şekil 3.1. deki alçak geçiren bir RC devresine benzeterek alçak frekans cevabını inceleyebiliriz. Şekildeki alçak geçiren RC devresinde gerilim kazancı

$$A_v = \frac{V_o}{V_s} = \frac{R_p}{R_p + R_s} \frac{1}{1 + \frac{1}{j\omega C_s(R_p + R_s)}} \quad \text{dir.}$$

Alt kesim frekansını

$$\omega_L = \frac{1}{(R_p + R_s)C_s} \quad \text{şeklinde gösterirsek}$$

$$A_v = \frac{R_p}{R_p + R_s} \frac{1}{1 - j \frac{\omega_L}{\omega}} \quad \text{şeklinde olur.}$$



Şekil 3.1 Alçak geçiren bir RC filtresi.

Ortak emetörlü bipolar transistörlü amplinin şekil 3.2.a da gösterilen eşdeğer devresinde 3 adet kapasite bulunduğundan gerilim ifadesi

$$A_v = A_{vo} \frac{1}{(1 - j \frac{\omega_{c1}}{\omega})(1 - j \frac{\omega_{c2}}{\omega})(1 - j \frac{\omega_{CE}}{\omega})} \quad \text{dir.}$$

w_{C1}, w_{C2}, w_{CE} devrede sadece C_1, C_2 veya C_E varken bulunacak köşe frekanslarıdır.

Bilindiği gibi alt kesim frekansı orta frekans kazancının $1/\sqrt{2}$ katına düştüğü frekans

değeridir. kazancın $1/\sqrt{2}$ katına düşmesi için kazanç ifadesinin paydasındaki ifade $\sqrt{2}$

ye eşit olmalıdır. Bilgisayar ile yapılan analizde iterasyon ile paydayı $\sqrt{2}$ yapacak olan w

değeri dolayısıyla w_L kolaylıkla bulunabilir.

Ancak kazanç ifadesindeki w_{C1}, w_{C2} ve w_{CE} değerlerinin bulunması gerekir. Bipolar transistörlü ve ortak emetörlü devre için C_1, C_2 ve C_E nin gördüğü eşdeğer dirençleri

şekil 2.3 deki basitleştirilmiş devreler yardımıyla kolayca bulunabilir.

C_1 in gördüğü direnç $(R_s + R_B // r_\pi)$

C_2 nin gördüğü direnç $R_C + R_L$

C_E nin gördüğü direnç ise $R_E // \frac{r_\pi + (R_s + R_B)}{\beta_o + 1}$ dir. Bu eşdeğer dirençler yardımıyla herbir kondansatöre ait köşe frekansının ifadesini yazarsak

$$w_{C1} = \frac{1}{[R_s + (R_B // r_\pi)] C_1}, \quad w_{C2} = \frac{1}{(R_C + R_L) C_2}, \quad w_{CE} = \frac{1}{\left[R_E // \frac{(r_\pi + R_s + R_B)}{\beta_o + 1} \right] C_E}$$

bulunur.

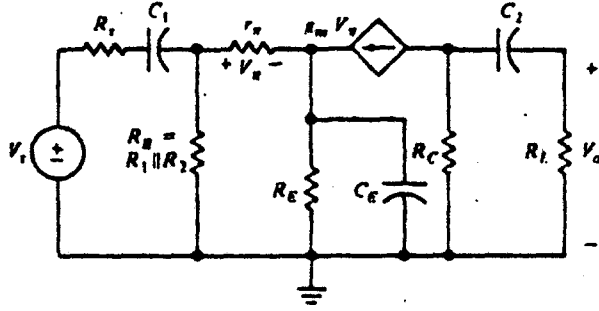
Şekil 2.2.b de gösterilen ortak bazlı ampli için de aynı metotla

$$w_{C1} = \frac{1}{(R_s + Z_i) C_1}, \quad w_{C2} = \frac{1}{(R_C + R_L) C_2}, \quad w_C = \frac{1}{R_B // [r_\pi + (\beta_o + 1)(R_s // R_E)] C_E}$$

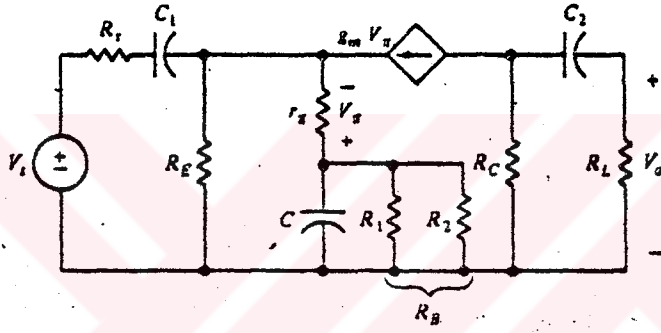
Ortak kollektörlü amplifikatörün şekil 2.2.c deki eşdeğer devresine bakarak benzer şekilde

$$w_{C1} = \frac{1}{(R_s + Z_i) C_1}, \quad w_{C2} = \frac{1}{(R_L + Z_o) C_2} \quad \text{ifadeleri bulunur. } Z_i \text{ ve } Z_o \text{ ortafrekans analizi}$$

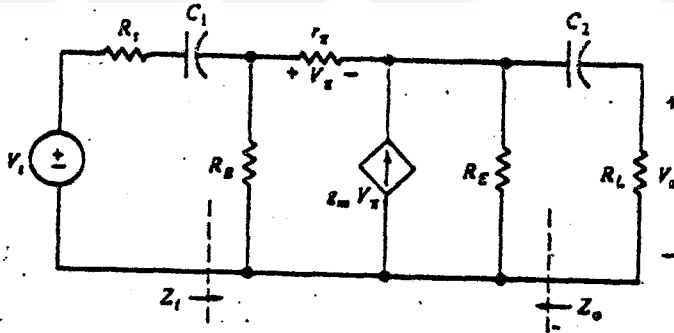
sırasında hesaplanan giriş ve çıkış empedanslarıdır.



(a)



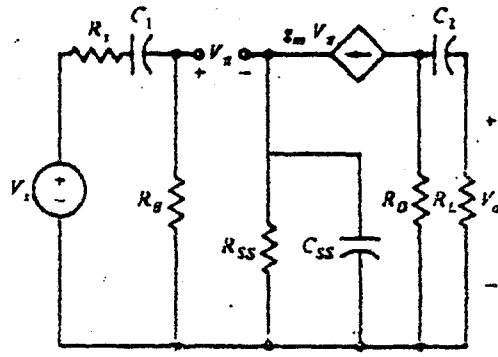
(b)



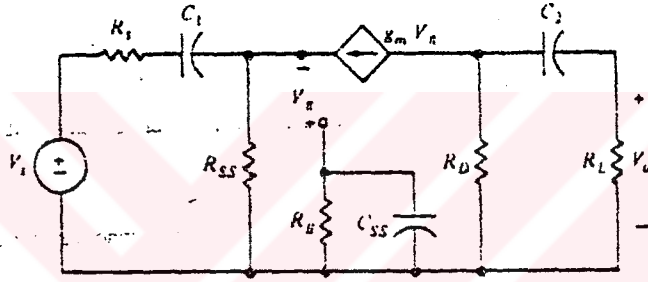
(c)

Şekil 3.2

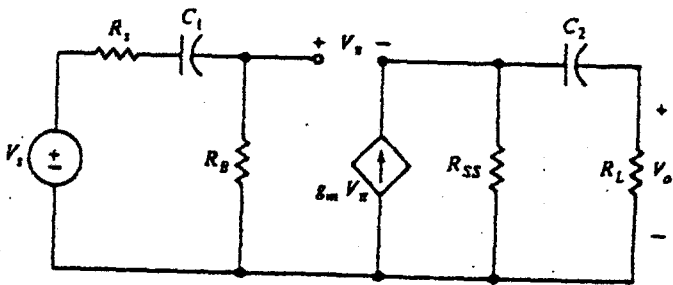
- Ortak emetörlü bipolar transistörlü amplinin alçak frekans eşdeğeri.
- Ortak kapılı bipolar transistörlü amplinin alçak frekans eşdeğeri.
- Ortak kollektörlü bipolar transistörlü amplinin alçak frekans eşdeğeri.



(a)



(b)



(c)

Şekil 3.3

- Ortak emetörlü fetli amplinin alçak frekans eşdeğeri.
- Ortak kapılı fetli amplinin alçak frekans eşdeğeri.
- Ortak kollektörlü fetli amplinin alçak frekans eşdeğeri.

3.2. FET li Amplifikatörün Alçak Frekans Cevabı

Bipolar transistörlü devrede olduğu gibi Fet li amplifikatörde de orta frekans analizi yapılırken kapasitelerin etkileri ihmal edilmiştir. Alçak frekanslarda bu kapasiteler etkili olmaktadır. Fet li amplifikatörün C1, C2 ve C_{ss} kapasitelerini de içeren orta frekans eşdeğer devresi ortak emetörlü, ortak bazlı ve ortak kollektörlü bağlantılar için şekil 3.3 de gösterilmiştir. Eşdeğer devrelerden

Ortak emetörlü bağlantı için;

$$w_{C1} = \frac{1}{(R_s + R_B)C1} , \quad w_{C2} = \frac{1}{(R_D + R_L)C2} \quad \text{ve} \quad w_{C2} = \frac{1}{[R_{ss} // (1/g_m)]C_{ss}}$$

Ortak bazlı bağlantı için;

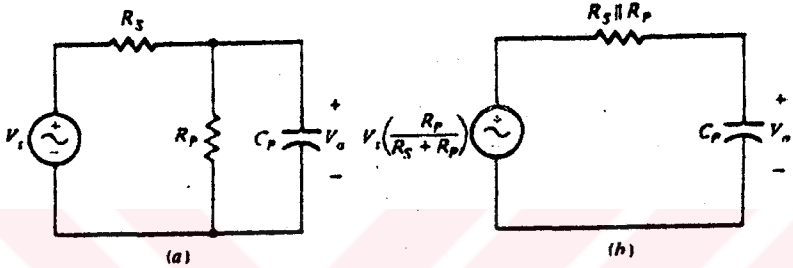
$$w_{C1} = \frac{1}{[R_s + (R_{ss} // 1/g_m)]C1} , \quad w_{C2} = \frac{1}{(R_D + R_L)C2} , \quad w_{C_{ss}} = \frac{1}{R_B C_{ss}}$$

Ortak kollektörlü bağlantı için de;

$$w_{C1} = \frac{1}{(R_s + R_B)C1} , \quad (Z_o = R_{ss} // 1/g_m) \quad \text{elde edilir.}$$

4. TEK KATLI AMPLİFİKATÖRÜN YÜKSEK FREKANSLARDAKİ ANALİZİ

Orta frekans analizi sırasında açık devre edilen C_{μ} ve C_{π} kapasiteleri yüksek frekanslarda etkili olmaktadır. Alçak frekans analizine benzer şekilde amplinin yüksek frekans eşdeğer devresini yüksek geçiren RC devresine benzetilerek analiz yapılacaktır. Şekil 4.1 de gösterilen yüksek geçiren RC devresinden



Şekil 4.1 Yüksek geçiren bir RC filtresi.

$$A_v = \frac{V_o}{V_s} = \frac{R_p}{R_p + R_s} \frac{1}{1 + j\omega C_p (R_s || R_p)} \text{ dir.}$$

$$\omega = \frac{1}{(R_s || R_p) C_p} \text{ konularak bu eşitlik düzenlenirse}$$

$$A_v = \frac{R_p}{R_p + R_s} \frac{1}{1 + j \frac{\omega}{\omega_H}} \text{ olur.}$$

4.1. Bipolar Transistörlü Amplifikatörün Yüksek Frekans Cevabı

Tek katlı amplifikatörün yüksek frekans eşdeğer devresinde iki adet kapasite sözkonusu olduğundan

$$A_v = A_{vo} \frac{1}{(1 + j \frac{\omega}{\omega_{cp}})(1 + j \frac{\omega}{\omega_{c\mu}})} \text{ şeklinde olur.}$$

Şekil 3.2. de gösterilen eşdeğer devrelerden ortak emetörlü ampli için,

$$C_{\pi} \text{ nin gördüğü direnç } R'_s = r_{\pi} // [r_X + (R_B // R_s)]$$

$$C_{\mu} \text{ nün gördüğü direnç } R_{\mu} = R'_L + R'_s(1 + g_m R'_L)$$

Ortak bazlı devrede,

$$R_{\pi} = r_{\pi} // \frac{R'_s + r_X}{1 + g_m R_s}, \quad R'_s = R_s // R_E$$

$$R_{\mu} = r_X + R'_L + \frac{r_X(\beta_0 R'_L - r_X)}{r_X + r_{\pi}(\beta_0 + 1)R_s}$$

Ortak kollektörlü bağlantıda

$$R_{\pi} = r_{\pi} // \frac{R'_s + R'_L}{1 + g_m R'_L}, \quad R'_s = r_X + (R_B // R_s), \quad R'_L = R_L // R_E$$

$$R_{\mu} = R'_s // [r_{\pi} + (\beta_0 + 1)R'_L]$$

Bu dirençlere karşılık gelen köşe frekansları ise

$$\omega_{c\pi} = \frac{1}{R_{\pi} C_{\pi}}, \quad \omega_{c\mu} = \frac{1}{R_{\mu} C_{\mu}} \text{ şeklindedir.}$$

4.2. FET li Amplifikatörün Yüksek Frekans Cevabı

Yukarıda takip edilen yola benzer şekilde şekil 4.3 de gösterilen fetli amplifikatörün eşdeğer devresinden ortak emetörlü devrede

$$C_{gs} \text{ in gördüğü direnç } R_{gs} = R_s // R_B$$

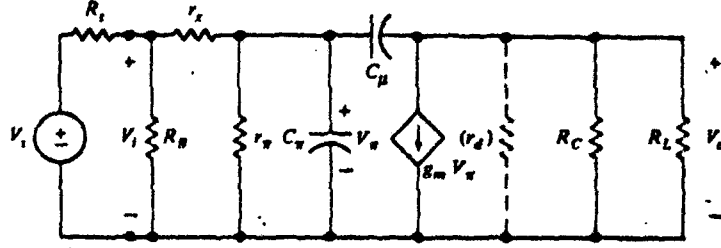
$$C_{gd} \text{ nin gördüğü direnç } R_{gd} = R'_L + (1 + g_m R'_L)R'_s$$

Ortak kapılı devrede

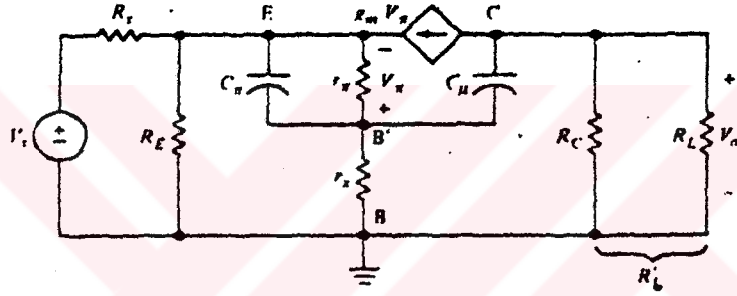
$$R_{gs} = R_s // R_{ss} // 1/g_m$$

$$R_{gd} = R'_L = R_D // R_L$$

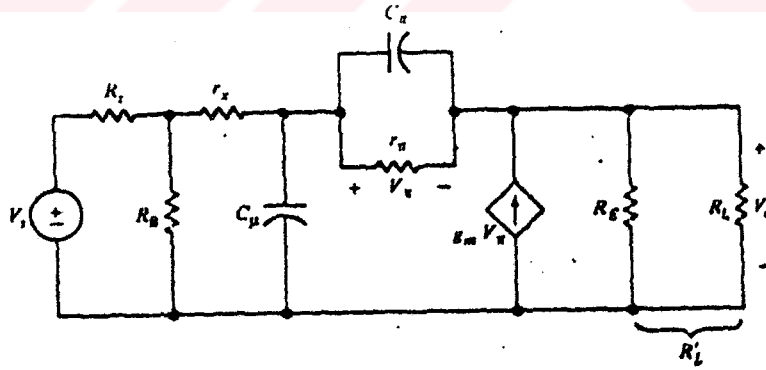
$$\text{Ortak kollektörlü devrede ise } R_{gs} = \frac{R'_s + R'_L}{1 + g_m R'_L}$$



(a)



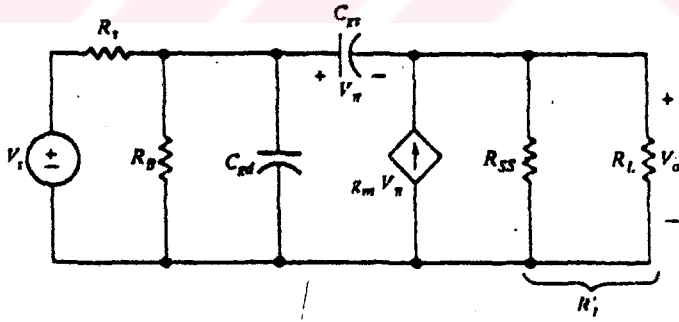
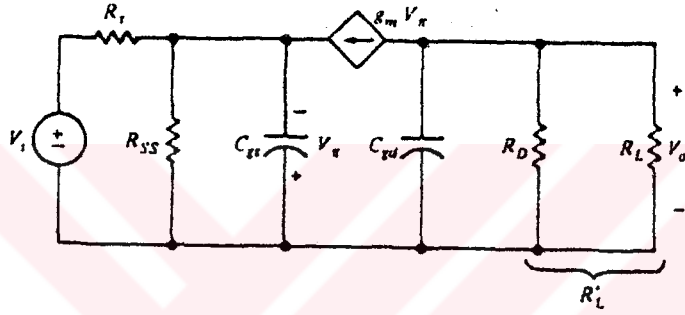
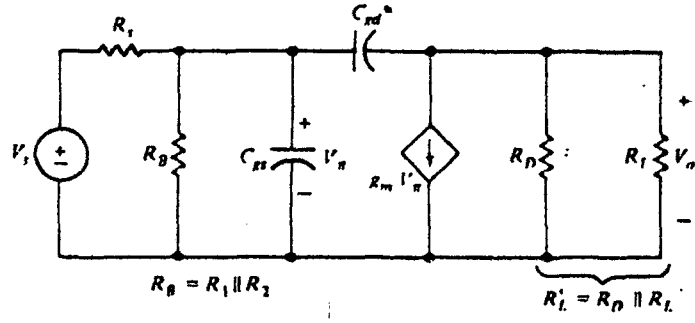
(b)



(c)

Şekil 4.2

- Ortak emetörlü bipolar transistörlü amplinin yüksek frekans eşdeğeri.
- Ortak kapılı bipolar transistörlü amplinin yüksek frekans eşdeğeri.
- Ortak kollektörlü bipolar transistörlü amplinin yüksek frekans eşdeğeri.



Şekil 4.3

- Ortak emetörlü fetli amplinin yüksek frekans eşdeğeri.
- Ortak kapılı fetli amplinin yüksek frekans eşdeğeri.
- Ortak kollektörlü fetli amplinin yüksek frekans eşdeğeri.

$$R_{gd} = R_s' = R_s // R_B$$

Bulunan R_{gs} ve R_{gd} dirençleri kullanılarak

$$w_{Cgs} = \frac{1}{R_{gs}C_{gs}} \quad , \quad w_{Cgd} = \frac{1}{R_{gd}C_{gd}} \quad \text{olur.}$$

5. TEK KATLI AMPLİFİKATÖRÜN TASARIMI

Önceki bölümlerde direnç ve kapasite değerleri belli olan tek katlı bir amplifikatörün ortak emetör ortak baz ve ortak kollektörlü bağlantılar için kazanç ve empedans değerleri ile alt ve üst kesim frekansları hesaplandı. Burada ise istenilen kazanç, giriş ve çıkış empedansı ile köşe frekanslarını elde etmek için direnç ve kapasitelerin hangi değerlerde olacağı hesaplanacaktır.

5.1. Tek Katlı Bipolar Transistörlü Amplifikatörün Tasarımı

Ortak emetörlü bağlantı için;

R_C yaklaşık Z_O a eşit alınabilir.

$$A_{vs} = g_m(R_C//R_L) \frac{Z_i}{R_s+Z_i} \text{ den}$$

$$g_m = \frac{A_{vs}}{(R_C//R_L) \frac{Z_i}{R_s+Z_i}} \text{ olur.}$$

$$r_\pi = \beta_o/g_m, \quad I_c = g_m/38.92$$

$$V_{cc} = R_C I_c + R_E I_E + V_{CE}$$

$$V_{cc} = V_{RC} + V_{RE} + V_{CE} \text{ dir.}$$

ilk tasarım için $V_{CE} = V_{RE}$ alınabilir. Böylece $V_{RE} = (V_{cc} - R_C I_c)/2$ olur.

$$R_E = V_{RE}/I_E, \quad I_b = I_c/\beta_{dc} \text{ olduğu gözönüne alınarak}$$

$$Z_i = r_\pi//R_B \quad \text{ve} \quad A_{vs} = g_m(R_C//R_L) \frac{Z_i}{R_s+Z_i} \text{ ifadelerinde yukarıdaki değerler}$$

kullanılıp alınan neticeler istenilen değerlerle karşılaştırılır. Z_i istenilen değerden küçük çıkıyorsa r_π veya R_B nin büyütülmesi gerekir. Eğer A_{vs} istenilen değerden küçük çıkıyorsa R_C direnci artırılarak A_{vs} istenilen değere yükseltilebilir. Sayısal bir örnek ile yukarıdaki metodun uygulamasını yapabiliriz.

Örneğin $A_{vs} \geq 100$, $Z_i \geq 3k\Omega$, $Z_o \geq 4k\Omega$, $R_s = 1k\Omega$ ve $R_L = 10k\Omega$ için ortak emetörlü bir amplifikatör tasarlayalım.

$R_c = Z_o = 4k\Omega$ seçelim.

$$g_m = \frac{100}{4//10 \frac{3}{3+1}} = 46.67 \text{ mmho}$$

$$\beta_o = \beta_{dc} = 120 \text{ ise}$$

$$r_\pi = 120/46.67 = 2.57k\Omega$$

$$I_c = g_m/38.92 = 1.2 \text{ mA}$$

$$V_{RE} = (V_{cc} - R_c I_c)/2, \quad V_{cc} = 20V \quad V_o = 0.65V \text{ alırsak}$$

$$V_{RE} = \frac{20 - 4 \cdot 1.12}{2} = 7.6$$

$$R_E = \frac{V_{RE}}{I_c} = 6.33k\Omega \quad I_b = 1.2/120 = 10\mu A$$

$$I_{R1} = 50I_B \text{ olduğu kabul edilirse}$$

$$R_1 = \frac{V_{cc} - V_{RE} - V_o}{I_{R1}} = \frac{20 - 7.6 - 0.5}{0.5} = 23.5k\Omega$$

$$R_2 = \frac{V_{RE} + V_o}{I_{R1} - I_b} = \frac{7.6 + 0.65}{0.5 - 0.01} = 16.83k\Omega$$

$$R_{th} = R_B = 23.5//16.83 = 9.8k\Omega$$

$$Z_i = r_\pi // R_B = 2.57//9.8 = 2.03k\Omega$$

görüldüğü gibi Z_i istenilen değerden küçüktür. Bu durumda R_c yi artırarak g_m i azaltabiliriz. Dolayısıyla r_π direncinin artmasını sağlayarak Z_i yi istenilen değere yükseltebiliriz.

$$R_c = 8k\Omega \text{ seçersek}$$

$$g_m = 30 \text{ mmho}, \quad r_\pi = 4k\Omega, \quad I_c = 0.77 \text{ mA}$$

$$R_E = 9k\Omega, \quad I_b = 6.4\mu A, \quad I_{R1} = 50 \cdot I_b = 0.321 \text{ mA}$$

$$R_1 = \frac{20-6.92-0.65}{0.321} = 38.72\text{k}\Omega$$

$$R_2 = \frac{6.92+0.65}{0.315} = 24.03\text{k}\Omega$$

$$R_B = R_1 // R_2 = 14.83\text{k}\Omega$$

$$Z_i = r_\pi // R_B = 3.15\text{k}\Omega$$

$$A_{vs} = -30(8/10) \frac{3.15}{1+3.15} = -101.2$$

$$Z_o \cong R_C = 8\text{k}\Omega$$

Bu değerler istenilen değerlere uygun olduğundan amplinin kazanç ve empedans hesabı tamamlanmış olur. Ortak baz ve ortak kolektörlü devreler için de benzer yaklaşımlar ile tasarım yapılabilir. Burada bir örnek de ortak baz bağlantılı ampli için verebiliriz.

$A_{vs} \geq 20$, $Z_i \leq 20\Omega$, $Z_o \geq 5\text{k}\Omega$, $R_s = 50\Omega$, $R_L = 5\text{k}\Omega$ özelliklerini sağlayacak ortak bazlı bir ampli tasarlayalım.

Yine $R_C \cong Z_o = 5\text{k}\Omega$ alalım.

$$g_m = \frac{20}{(5/15) \frac{20}{20+50}} = 28 \text{ mmho}$$

$\beta_o = \beta_{dc} = 120$ verilmiş olsun.

$$r_\pi = \frac{120}{28} = 4.29 \text{ k}\Omega$$

$$I_C = \frac{28}{38.92} = 0.72 \text{ mA}$$

$$V_{RE} = \frac{20-5-0.72}{2} = 8.2$$

$$R_E = \frac{8.2}{0.72} = 11.39 \text{ k}\Omega, \quad I_b = \frac{0.72}{120} = 6 \text{ }\mu\text{A}$$

$$I_{R1} = 50I_b = 0.3 \text{ mA},$$

$$R_1 = \frac{20-8.2-0.65}{0.3} \cong 38 \text{ k}\Omega, \quad R_2 = \frac{8.2+0.65}{0.294} \cong 30 \text{ k}\Omega$$

$$R_{th} = R_B = 38//30 = 16.76 \text{ k}\Omega$$

$$Z_i = R_E // r_{\pi} // \frac{1}{g_m} = 11.4 // 4.29 // \frac{1}{28} = 35 \Omega$$

Görüldüğü gibi Z_i istenilen değerden büyüktür. Bu durumda g_m i artırmak veya R_E yi azaltmak gerekir.

$\frac{1}{g_m} = \frac{1}{28} = 0.36 \text{ k}\Omega > Z_i$ dir. Bu durumda R_E nin azaltılması Z_i ye fazla tesir etmeyecektir. O halde R_C azaltılarak $2 \text{ k}\Omega$ seçilirse

$$g_m = 49 \text{ mmho}, \quad r_{\pi} = \frac{120}{49} = 2.45 \text{ k}\Omega$$

$$I_C = \frac{49}{38.92} = 1.23 \text{ mA},$$

$$V_{RE} = \frac{20-2 \cdot 1.23}{2} = 8.77, \quad R_E = 7.13 \text{ k}\Omega, \quad I_b = \frac{1.23}{120} = 10.2 \mu\text{A}$$

$$I_{R1} = 50I_b = 0.513 \text{ mA}$$

$$R_1 = \frac{20-8.77-0.65}{0.513} = 20.62 \text{ k}\Omega, \quad R_2 = \frac{8.77+0.65}{0.503} = 18.73 \text{ k}\Omega$$

$$R_B = 9.81 \text{ k}\Omega$$

$$Z_i = R_E // r_{\pi} // \frac{1}{g_m} = 7.13 // 2.45 // \frac{1}{49} \cong 20 \Omega$$

$$A_{vs} = 49(2//5) \frac{20}{20+50} = 20$$

bulunan bu sonuçlar uygun olduğundan tasarım tamamlanmış olur. İkinci aşamada istenilen köşe frekansını elde edebilmek için kapasitelerin hesabına geçebiliriz.

Amplifikatörün alçak frekans analizi yapılırken devrede bulunan üç kapasiteden herbiri için bir köşe frekansı bulunmuştur. Ortak emetör, ortak baz ve ortak kollektör bağlantılı amplifikatörlerin herbirinde bir kapasiteye ait frekans daha etkin olmaktadır. Tasarımın başlangıcında alt kesim frekansının hesabında en çok etkili olan köşe frekansının alt kesim frekansına eşit diğerlerinin ise buna göre 5-10 kat daha küçük olduğunu kabul edebiliriz. Ortak emetörlü devrede w_{CE} frekansı en çok etkili olduğu için

$w_{CE} = w_L$, $w_{C1} = w_L/10$, $w_{C2} = w_L/10$ kabul edelim. Bölüm 3.1 deki ifadeler yardımıyla

$$w_{CE} = \frac{1}{R_{th} C_E} \Rightarrow C_E = \frac{1}{R_{th} w_L}$$

$$w_{C1} = \frac{1}{[R_s + (R_B // r_\pi)] C_1} \Rightarrow C_1 = \frac{1}{[R_s + (R_B // r_\pi)] \frac{w_L}{10}}$$

$$w_{C2} = \frac{1}{(R_C + R_L) C_2} \Rightarrow C_2 = \frac{1}{(R_C + R_L) \frac{w_L}{10}}$$

Elde edilen değerlerle kontrol yapılır. Ortak emetör bağlantılı devrede w_{CE} köşe frekansı, alt kesim frekansının tesbitinde en etkin köşe frekansı olduğundan C_E nin değeri uygun yönde değiştirilerek istenilen w_L değerine ulaşılır. Yukarıda ortak emetörlü devre için verdiğimiz örnekte $w_L \leq 1000$ rad/s olması istenmiş olsun. Alt kesim frekansı orta frekans kazancının $\frac{1}{\sqrt{2}}$ katına düştüğü değer olduğuna göre,

$$A_v = A_{vo} \frac{1}{(1 - j \frac{w_{C1}}{w})(1 - j \frac{w_{C2}}{w})(1 - j \frac{w_{CE}}{w})} \quad \text{ifadesinde paydanın genliğinin } \sqrt{2} \text{ olması}$$

gerekir. yukarıdaki ifadenin paydasında w_{C1} , w_{C2} , w_{CE} yerine konulup paydayı $\sqrt{2}$ yapacak w değeri sayısal analiz yollarından biri kullanılarak 1018 rad/s bulunur. Halbuki bizden $w \leq 1000$ rad/s olması istenmişti. Bu durumda w_{CE} yi bir miktar küçültmek gerekir. $w_{CE} = 950$ rad/s alıp iterasyon yenilenirse $w=970$ rad/s bulunur. Bu ise uygun bir değer olduğundan bu değerler yardımıyla C_1 , C_2 ve C_E yi hesaplayabiliriz.

$$C_1 = \frac{10^{-3}}{[1 + (14.83/\omega)]100} = 2.4 \mu F$$

$$C_2 = \frac{10^{-3}}{(8+10)100} = 0.56\mu F$$

$$C_E = \frac{10^{-3}}{\left[9 // \frac{4+(1//14.83)}{121}\right]950} = 25.9\mu F \quad \text{bulunur. Bundan sonra amplifikatörün üst}$$

kesim frekansının hesabına geçebiliriz.

Amplifikatörün yüksek frekans analizinde görüldüğü gibi yüksek frekans eşdeğer devresinde C_p ve C_μ olmak üzere iki adet kapasite bulunmaktadır. Dolayısıyla kazanç ifadesi

$$A_v = A_{vo} \frac{1}{(1 + j \frac{w}{w_{C\pi}})(1 + j \frac{w}{w_{C\mu}})} \quad \text{şeklinde olur.}$$

Burada da yapılacak işlem alçak frekansda olduğu gibi paydayı $\sqrt{2}$ yapan w değerini bulmak olacaktır. Kataloglarda belirli bir V_{cb} değeri için C_μ nün değeri verilir. Buradan $C_{\mu 0}$ hesaplanarak V_{cb} nin farklı değerleri için C_μ hesaplanabilir. $w > w_H$ olması istendiğine ve C_μ de belirlendiğine göre C_π ye çeşitli değerler vererek paydayı $\sqrt{2}$ yapan C_π değeri bulunabilir. Yine ortak emetörlü devre için verdiğimiz birinci örneğe dönelim. $w_H \geq 5\text{MRad/s}$ olması istenilmiş olsun. Kullanılan transistör için katalogdan $V_{cb} = 4.35\text{V}$ için $C_\mu = 2 \text{ pF}$ verilmiş ise:

$$C_\mu = \frac{C_{\mu 0}}{(1 + \frac{V_{cb}}{V_{bi}})^N}, \quad \text{silyum transistörlerde } V_{bi} = 0.8\text{V}, \quad n \text{ ise } 1/2 \text{ ile } 1/3$$

arasındadır.

$$C_{\mu 0} = 2 \cdot 10^{-12} (1 + \frac{4.35}{0.8})^{\frac{1}{2}} = 5\text{pF} \quad \text{bulunur.}$$

örneğimizde

$$V_{cb} = V_{CE} - V_{BE} = 6.92 - 0.65 = 6.27 \text{ V}$$

$$C_\mu = \frac{5 \cdot 10^{-12}}{(1 + \frac{6.27}{0.8})^{\frac{1}{2}}} = 1.68 \text{ pF bulunur.}$$

$$R'_L = (R_C // R_L) = 8 // 10 = 4.44\text{k}\Omega$$

$$R'_S = r_\pi / [r_x + (R_B // R_S)] , \quad r_x = 20\Omega \text{ için;}$$

$$R'_S = 4 / [0.02 + (14.83 // 1)] = 0.77k\Omega$$

$$R_\mu = R'_{I_1} + R'_S(1 + g_m R'_{I_1}) = 4.44 + 0.77(1 + 30 \cdot 4.44) = 107.77k\Omega$$

$$w_{C_\mu} = \frac{1}{R_\mu C_\mu} = \frac{10^{-3}}{107.77 \cdot 1.68 \cdot 10^{-12}} = 5. \text{ Mrad/s}$$

kazanç ifadesinde $w = w_H$ ve $w_{C_\mu} = 5.52 \text{ Mrad/s}$ degerleri yerleştirilerek paydayı $\sqrt{2}$ yapmak için iterasyon yapılırsa $w_{C_\pi} = 15.8 \text{ Mrad/s}$ bulunur. Buradan

$$C_\pi = \frac{1}{R_\pi w_{C_\pi}} , \quad R_\pi = R'_S = 0.77k\Omega \text{ idi.}$$

$$C_\pi = \frac{10^{-9}}{0.77 \cdot 15.8} = 82 \text{ pF bulunur.}$$

Bipolar ransistörler için $C_\pi - C_\mu$ arasındaki bağıntıyı gösteren

$$C_\pi = \frac{g_m}{W_T} - C_\mu \text{ ifadesinden } W_T = 358 \text{ Mrad/s bulunur.}$$

Öyle ise $w_H \geq 5 \text{ Mrad/s}$ elde edebilmek için kullanılan transistörde $W_T \geq 358 \text{ Mrad/s}$ olmalıdır.

5.2. Tek Katlı Fetli Amplifikatörün Tasarımı

$$A_{vs} \geq 10, \quad R_S = 1k\Omega, \quad R_L = 10k\Omega, \quad v_p = -3, \quad I_{DSS} = 10mA, \quad Z_i \geq 200k\Omega,$$

$Z_o \geq 10k\Omega$ için ortak emetör bağlantılı bir amplifikatör tasarlayalım.

$$A_{vs} = g_m(R_D // R_L) \frac{Z_i}{R_S + Z_i} \text{ idi.}$$

Ortak emetörlü devre için $R_D = Z_o = 10k\Omega$ alınabilir.

$$g_m = \frac{A_{vs}}{(R_D // R_L) \frac{Z_i}{R_S + Z_i}} = \frac{10}{(10 // 5) \frac{200}{200+1}} = 3 \text{ mmho}$$

$$g_m = \frac{2I_{DSS}}{V_p^2}(V_{GS} - V_p) \text{ olduğuna göre ;}$$

$$3 = \frac{2 \cdot 10}{(-3)^2}(V_{GS} + 2) \quad , \quad V_{GS} = -0.65V$$

$$I_d = \frac{I_{DSS}}{V_p^2}(V_{GS} - V_p)^2 = \frac{10}{(-3)^2}(-0.65 + 2)^2$$

$$I_d = 2mA$$

$$R_{SS}I_d + V_{GS} = V_G = \frac{R_2}{R_1 + R_2}V_{DD} \text{ eşitliğini kullanarak}$$

$$R_{SS} = \frac{V_G - V_{GS}}{I_d} \text{ yazılabilir. Ortak emetörlü ve fetli amplifikatör için } R_2 = Z_i ,$$

$$R_1 = \infty \text{ alınabilir. Bu durumda;}$$

$$R_{SS} = \frac{-V_{GS}}{I_d} = \frac{0.65}{2} = 0.325k\Omega \text{ bulunur.}$$

Bu amplifikatör için $w_L \leq 500 \text{ rad/s}$ ve $w_H \geq 20 \text{ MRad/s}$ için uygulama yapalım.

w_L alt kesim frekansını bulmak için devredeki C_1, C_2 ve C_{SS} kapasitelerini gözönüne almak gerekir. Bipolar transistörlü amplitin tasarımında uygulanan aynı yöntemle

$$A_v = A_{vo} \frac{1}{(1 - j\frac{wC_1}{w})(1 - j\frac{wC_2}{w})(1 - j\frac{wC_{SS}}{w})}$$

ifadesinde payda $\sqrt{2}$ ye eşitlenip w nın istenilen değere ulaşması için $w_{C_{SS}}$ e uygun

değerler verilip gerekli hesaplar yapılır.

$w_{C_1} = 50 \text{ rad/s}$, $w_{C_2} = 50 \text{ rad/s}$ ve $w_{C_{SS}} = 500 \text{ rad/s}$ alınarak iterasyon yapılırsa paydayı $\sqrt{2}$ yapmak için $w = 510 \text{ rad/s}$ bulunur. Bu sonuç istenilen değerden büyüktür.

Bu durumda $w_{C_{SS}}$ biraz küçültülerek 450 rad/s alınırsa $w = 461 \text{ rad/s}$ için payda $\sqrt{2}$ ye eşitlenmiş olur. $w = 461 \text{ rad/s}$ uygun bir değer olduğundan kapasitelerin hesabına geçebiliriz.

Bölüm 2.2 deki ifadeler yardımıyla

$$C_1 = \frac{1}{(R_S + R_B)w_{C1}} = \frac{10^{-3}}{(1 + 200) \cdot 50} = 0.1 \mu F$$

$$C_2 = \frac{1}{(R_D + R_L)w_{C2}} = \frac{10^{-3}}{(10 + 10) \cdot 50} = 1 \mu F$$

$$C_{ss} = \frac{1}{(R_{ss} // \frac{1}{g_m})w_{C_{ss}}} = \frac{10^{-3}}{(0.325 // \frac{1}{3}) \cdot 450} = 13.5 \mu F$$

w_H üst kesim frekansının istenilen değerde olması için $w_{C_{gs}}$ ve $w_{C_{gd}}$ değerlerini

bulmak gerekir. Katalogdan $C_{gd} = 2 \text{ pF}$ olduğu kabul edilirse bölüm 3.2 deki ifadeleri kullanarak

$$R_{gd} = R'_L + (1 + g_m R'_L) R'_S,$$

$$R'_L = R_D // R_L = 10 // 10 = 5 \text{ k}\Omega, \quad R'_S = R_S // R_B = 1 // 200 = 0.995 \text{ k}\Omega$$

$$R_{gd} = 5 + (1 + 3 \cdot 5) \cdot 0.995 = 20.92 \text{ k}\Omega$$

$$w_{C_{gd}} = \frac{1}{R_{gd} C_{gd}} = \frac{1}{20.92 \cdot 2 \cdot 10^{-12}} = 23.4 \text{ MRad/s}$$

konularak $w_{C_{gs}} = 47.5 \text{ MRad/s}$ bulunur.

$$w_{C_{gs}} = \frac{1}{R_{gs} C_{gs}} \text{ ifadesinden de}$$

$$C_{gs} = \frac{1}{R_{gs} w_{C_{gs}}} = \frac{10^{-9}}{0.995 \cdot 47.5} = 21 \text{ pF bulunur. Böylece amplifikatörün tasarımı}$$

tamamlanmış olur.

6. TEK KATLI BİR AMPLİFİKATÖRÜN ANALİZ VE TASARIMINI YAPAN BİLGİSAYAR PROGRAMI

```
{ ANA PROGRAM }
uses crt,graph,ampli;
var
dos          :file of real;
doskont,j1,j2 :Byte;
ch1,ch2      :Char;
x1,x2        :integer;
da:string[12];
s:string[4];
str1:string[255];
ch:char;
r11,r21,r1,rc,rd,re,rs,r0,rbb,vcc,v0,bdc,bdcmin,bdcm,b0,
b0min,b0m,ice0,avi,avs,avs0,zi,zi0,ai,ap,z0,z00,rb,gm,
rp,rth,idss,vp,vdd,rs s,kk1,vt,w,wcl,wc2,wce,wz,wcp,
wcmu,wcss,wc,wl,c1,c2,c3,ce,css,rgs,rgd,tgs,tgd,t,wh,
tp,tm,cgs,cgd,cp,cmu,cmu0,rx,vce,vcb,vre,vth,vds,vgs,id,
z,ic,rcl,i b,irl,kok,keb,kebl,keb2:real;
i,j,k,kl:integer;
const
ins:byte=0;kayit:byte=0;ilkchar:byte=0;dosk:byte=0;iptal
:byte=0;
dabuf:string[12]='';
d:byte=1;f:Byte=1;f0:Byte=0;e:Byte=1;b:Byte=1;c:Byte=1;
kk:real=50;il:Byte = 1;i2:Byte = 1;j3:Byte = 1;
rk:Byte = 0;r1:Byte = 0;r2:Byte = 7;r3:Byte = 7;
r4:Byte = 0;r5:Byte = 7;r6:Byte = 0;r7:Byte = 7;
r8:Byte = 0;km:byte = 0;fm:byte = 0;dm:byte = 0;
dmm:byte= 0;em:byte = 0;bm:byte = 0;
Label oku,okuu,son;
PROCEDURE Dcbipolar;
var
p,q,r,s,y,z:real;
BEGIN
rb:=paralel(r11,r21);rth:=rb+rbb;vth:=vcc*r21/(r11+r21);
if r0=-1 then begin p:=1;s:=0;end
else begin
p:=1-re/bdc/r0;s:=- (rth+re)/bdc/r0;end;
q:=rc+re*(bdc+1)/bdc;r:=vcc+re*ice0/bdc;
y:=(rth+re)/bdc+re;z:=(rth+re)*ice0/bdc-v0+vth;
vce:=(y*r-q*z)/(p*y-q*s);vcb:=vce-v0;
ic:=(s*r-p*z)/(s*q-p*y);
gm:=38.92*ic;
rp:=b0/gm;
if f=1 then begin
window(47,3,79,23);
clrscr;
if (vce<v0) then
begin
textcolor(15+blink);
writeln('Transistör doyma bölgesi dışında');
textcolor(7);
end;
```

```

writeln('rth=',rth:4:2,' kΩ');
writeln('vth=',vth:4:2,' V');
writeln('Vce=',vce:4:2,' V');
writeln('Ic =',ic:4:2,' mA');
writeln('gm =',gm:4:2,' mmho');
writeln('r',chr(210),' =',rp:4:2,' kΩ');
end;
END;
PROCEDURE Dcfet;
var
p,q,r:real;
BEGIN
if b=3 then rd:=0;
if rk<>1 then rb:=paralel(r11,r21);
if rk=1 then rb:=r21;
p:=rss*idss/sqr(vp);q:=1-2*rss*idss/vp;
if rk<>1 then r:=(rss*idss-r21*vdd)/(r11+r21)
else r:=rss*idss;
if rss=0 then vgs:=0 else
begin
vgs:=(-q+sqr(q*q-4*p*r))/2/p;
if (vgs>0) or (vgs<vp) then vgs:=(-q-sqr(q*q-
4*p*r))/2/p;
end;
id:=idss*(vgs-vp)*(vgs-vp)/(vp*vp);
vds:=vdd-(rd+rss)*id;
vgs:=vp-vp*sqr(id/idss);
gm:=2*idss*(vgs-vp)/sqr(vp);
if f=1 then begin
window(50,4,79,23);
clrscr;
if ((vgs>0) or (vgs<vp) or (vds<vgs-vp)) and (k=1) then
begin
textcolor(15+blink);
writeln('FET doyma
bölgəsi disinda');
textcolor(7);
end;

writeln('ID =',id:4:2,' mA');
writeln('VDS=',vds:4:2,' V');
writeln('VGS=',vgs:4:2,' V');
writeln('gm =',gm:4:2,' mmho');
end;
END;
PROCEDURE dcarttirilmis;
var
vg,p,q,y:real;
BEGIN
if b=3 then rd:=0;
if rk<>1 then rb:=paralel(r11,r21);
if rk=1 then rb:=r11;
p:=rss*kk1/2;
q:=1-rss*kk1*vt;
if rk=1 then y:=rss*kk1*sqr(vt)/2-vdd

```

```

        else y:=rss*kk1*sqr(vt)/2-vdd*r21/(r11+r21);
vgs:=(-q+sqr(q*q-4*p*y))/2/p;
if (vgs<vt) or (vgs<0) then
    vgs:=(-q-sqr(q*q-4*p*y))/2/p;
id:=kk1*sqr(vgs-vt)/2;
vds:=vdd-(rd+rss)*id;
gm:=kk1*(vgs-vt);
if f=1 then begin
window(50,4,79,23);
clrscr;
if ((vgs<vt) or (vgs<0) or (vds<vgs-vt)) and (k=1) then
begin
    textcolor(15+blink);
    writeln('FET doyma bölgesi disinda');
    textcolor(7);
    end;
writeln('ID=',id:4:2,' mA');
writeln('VDS=',vds:4:2,' V');
writeln('VGS=',vgs:4:2,' V');
writeln('gm=',gm:4:2,' mmho');
    end;
END;
PROCEDURE ofbipemt;
BEGIN
dcbipolar;
zi:=paralel(rp,rb);
if rd=-1 then begin z0:=rc;avi:=-gm*paralel(rc,r1);end
    else begin z0:=paralel(rc,rd);
        avi:=-gm*paralel(paralel(rd,rc),r1);
        end;
av:=avi*zi/(rs+zi);
ai:=avi*zi/r1;
ap:=avi*avi*zi/r1;
if f=1 then begin
writeln('Avi=',avi:4:2);
writeln('Avs=',avs:4:2);
writeln('Zi =',zi:4:2,' kΩ');
writeln('Ai =',ai:4:2);
writeln('Ap =',ap:4:2);
writeln('Z0 =',z0:4:2,' kΩ');
end;
END;
PROCEDURE ofbipbaz;
BEGIN
dcbipolar;
if rd=-1 then begin
avi:=gm*paralel(rc,r1);zi:=paralel(paralel(re,rp),1/gm);
z0:=rc;
        end
        else begin
avi:=(gm+(1-
gm*paralel(rc,r1))/(rd+paralel(rc,r1)))*paralel(rc,r1);
zi:=paralel(paralel((rd+paralel(rc,r1))/
(rd+2*paralel(rc ,r1))*(1/gm),rp),re);

```

```

z0:=rc-(1-gm*paralel(rc,r1))/
(gm*(rd+paralel(rc,r1))) *rc;
      end;
avs:=avi*zi/(rs+zi);
ai:=avi*zi/rl;
ap:=avi*avi*zi/rl;
if f=1 then Begin
writeln('Avi=',avi:4:2);
writeln('Avs=',avs:4:2);
writeln('Zi =',zi*1000:4:2,'  $\Omega$ ');
writeln('Ai =',ai:4:2);
writeln('Ap =',ap:4:2);
writeln('Z0 =',z0:4:2,'  $k\Omega$ ');
      End;
END;
PROCEDURE ofbipcol;
BEGIN
dcbipolar;
if rd=-1 then begin
zi:=rp+(1+b0)*paralel(re,r1);zi:=paralel(zi,rb);
z0:=paralel(re,(rp+paralel(rb,rs))/(b0+1));
avi:=(1+b0)*paralel(re,r1)/(rp+(1+b0)*paralel(re,r1));
      end
      else begin
zi:=rp+(1+b0)*paralel(rd,paralel(re,r1));
zi:=paralel(zi,rb);
z0:=paralel(rd,paralel(re,(rp+paralel(rb,rs))/(b0+1)));
avi:=(1+b0)*paralel(rd,paralel(re,r1))/
(rp+(1+b0)*paralel(rd,paralel(re,r1)));
      end;

avs:=avi*zi/(rs+zi);
ai:=avi*zi/rl;
ap:=avi*ai;
if f=1 then begin
writeln('Avi=',avi:4:2);
writeln('Avs=',avs:4:2);
writeln('Zi =',zi:4:2,'  $k\Omega$ ');
writeln('Ai =',ai:4:2);
writeln('Ap =',ap:4:2);
writeln('Z0 =',z0*1000:4:2,'  $\Omega$ ');
      end;
END;
PROCEDURE offetemt;
BEGIN
avi:=-gm*paralel(rd,r1);
if (rk<>1) or (b=2) then zi:=paralel(r11,r21)
      else if e=2 then zi:=r21

else zi:=r11;
avs:=avi*zi/(rs+zi);
ai:=avi*zi/rl;
ap:=avi*ai;
z0:=rd;

```

```

if f=1 then begin
writeln('Avi=',avi:4:2);
writeln('Avs=',avs:4:2);
writeln('Zi =',zi:4:2,' kΩ');
writeln('Ai =',ai:4:2);
writeln('Ap =',ap:4:2);
writeln('Z0 =',z0:4:2,' kΩ');
end;
END;
PROCEDURE offetbaz;
BEGIN
avi:=gm*paralel(rd,rl);
zi:=paralel(rss,(1/gm));
avs:=avi*zi/(rs+zi);
ai:=avi*zi/rl;
ap:=avi*ai;
z0:=rd;
if f=1 then begin
writeln('Avi=',avi:4:2);
writeln('Avs=',avs:4:2);
writeln('Zi =',zi*1000:4:2,' Ω');
writeln('Ai =',ai:4:2);
writeln('Ap =',ap:4:2);
writeln('Z0 =',z0:4:2,' kΩ');
end;
END;
PROCEDURE offetcol;
BEGIN
avi:=gm*paralel(rss,rl)/(1+gm*paralel(rss,rl));
zi:=rb;
avs:=avi*zi/(zi+rs);
ai:=avi*zi/rl;
ap:=avi*ai;
z0:=paralel(rss,(1/gm));
if f=1 then begin
writeln('Avi=',avi:4:2);
writeln('Avs=',avs:4:2);
writeln('Zi =',zi:4:2,' kΩ');
writeln('Ai =',ai:4:2);
writeln('Ap =',ap:4:2);
writeln('Z0 =',z0*1000:4:2,' Ω');
end;
END;
function fonk(w:real):real;
var
im,sanal,rthce,rpi,rmu:real;
s:string[30];
BEGIN
case f of
2:case b of
1:Begin
if e=1 then begin
wc1:=1E-03/(rs+paralel(rb,rp))/c1;
wc2:=1E-03/(rc+rl)/c2;

```

```

rthce:=(rp+paralel(rs,rb))/(b0+1);
rthce:=paralel(rthce,re);
wce:=1E-03/rthce/ce;
wz:=1E-03/re/ce;
    end
    else begin
wcl:=1E-03/(rs+rb)/c1;
wc2:=1E-03/(rd+rl)/c2;
if (rss<>0) and (css<>-1) then
wce:=1E-03/paralel(rss,(1/gm))/css
    else wce:=0;
    end;
End;
2:Begin
if e=1 then begin
wcl:=1E-03/(zi+rs)/c1;
wc2:=1E-03/(rc+rl)/c2;
wce:=1E-03/
paralel(rb,(rp+(b0+1)*paralel(rs,re)))/c3;
    end
    else begin
wcl:=1E-03/(rs+zi)/c1;
wc2:=1E-03/(rd+rl)/c2;
if rb<>0 then wce:=1E-03/rb/c3
    else wce:=0;
    end;
End;
3:Begin
if e=1 then begin
wcl:=1E-03/(rs+zi)/c1;
wc2:=1E-03/(rl+z0)/c2;
    end
    else begin
wcl:=1E-04/(rs+rb)/c1;
wc2:=1E-03/(rl+z0)/c2;
    end;
End;
End;
3:case b of
1:Begin
if e=1 then begin
rthce:=paralel(rp,(rx+paralel(rb,rs)));
wcp:=1E-03/rthce/cp;
wcmu:=1E-
03/(paralel(rc,rl)+rthce*(1+gm*paralel(rc,
rl)))/cmu;
    end
    else begin
wcp:=1E-03/paralel(rs,rb)/cgs;
wcmu:=1E-03/(paralel(rd,rl)+
(1+gm*paralel(rd,rl))*paralel(rs,rb))/cgd;
    end;
End;
2:Begin

```

```

    if e=1 then begin
      rthce:=paralel(rp,(paralel(rs,re)+rx)/
        (1+g m*paralel(rs,re)));
      wcp:=1E-03/rthce/cp;
      rthce:=rx+paralel(rc,rl)+
        (rx*(b0*paralel(r c,rl)-rx)/
        (rx+rp+(b0+1)*paralel(rs,re)));
      wcmu:=1E-03/rthce/cmu;
      end
    else begin
      wcp:=1E-03/paralel(rs,zi)/cgs;
      wcmu:=1E-03/paralel(rd,rl)/cgd;
      end;
    End;
  3:Begin
    if e=1 then begin
      rthce:=paralel((rx+paralel(rb,rs)),
        (rp+(b0 +1)*paralel(re,rl)));
      wcmu:=1E-03/rthce/cmu;
      rthce:=paralel(rp,(((rx+paralel(rb,rs))+
        pa ralel(re,rl))/(1+gm*paralel(re,rl))));
      wcp:=1E-03/rthce/cp;
      end
    else begin
      wcp:=1E-03/paralel(rs,rb)/cgd;
      wcmu:=1E-03/
        ((paralel(rs,rb)+paralel(rl,rss))/
        (1+gm *paralel(rl,rss)))/cgs;
      end;
    End;
  end;
End;
if f=2 then begin
  if b=1 then
  begin
    im:=1-(wc1*wc2+wc2*wce+wc1*wce+wc2*wz+wc1*wz+wce*wz)/w/w
      -wc1*wc2*wce*wz/w/w/w/w;
    sanal:=(wc1*wc2*wce+wc1*wc2*wz+wc2*wce*wz+wc1*wce*wz)/w/
      w/w-(wc1+wc2+wce+wz)/w;
  end;
  if b=2 then
  begin
    im:=1-(wc1*wc2+wc1*wce+wc2*wce)/w/w;
    sanal:=(wc1*wc2*wce/w/w/w)-(wc1+wc2+wce)/w;
  end;
  if b=3 then
  begin
    im:=1-wc1*wc2/w/w;
    sanal:=- (wc1+wc2)/w;
  end;
  end;
if f=3 then begin
  im:=1-w*w/wcp/wcmu;
  sanal:=w/wcp+w/wcmu;

```

```

        end;
fonk:=sqrt(im*im+sanal*sanal);
END;
PROCEDURE whesabi;
var w1,w2:real;
BEGIN
w1:=0;w2:=1E+15;
repeat
if fonk((w1+w2)/2) > kok then
case f of
2:begin w1:=(w1+w2)/2;w:=w1;end;
3:begin w2:=(w1+w2)/2;w:=w2;end;
end
else
case f of
2:begin w2:=(w1+w2)/2;w:=w2;end;
3:begin w1:=(w1+w2)/2;w:=w1;end;
end;
until (abs(fonk(w)-kok)<=1E-06) or ((abs(w1-w2))<=1);
if f=2 then w1:=w else wh:=w;
END;
PROCEDURE afbip;
var
grDriver : Integer;
grMode   : Integer;
ErrCode   : Integer;
i,j,bry,us,wk,xa,xb,ap,alm,ulm:integer;
wx,wd,br:real;
st:string[20];
say:string[1];
sayi:string[5];
PROCEDURE aps(x1,y1,x,y,z:integer);
BEGIN
str(z,st);
if z>=1 then outTextxy(x1,y1,'10');
if z>1 then outTextxy(x,y,st);
if z=0 then outTextxy(x1,y1,'1');
End;
begin
whesabi;
grDriver := Detect;
InitGraph(grDriver,grMode,'');
ErrCode := GraphResult;
if ErrCode = grOk then
begin
avs:=abs(avs);
str(trunc(20*ln(avs)/ln(10)),st);
str(abs(trunc(100*Frac(20*ln(avs)/ln(10))))),sayi);
outTextxy(510,250,'Avso='+st+'.'+sayi+'dB');
alm:=400;ulm:=50;
br:=(alm-ulm)/9;
bry:=abs(trunc(20*ln(avs)/ln(10)/6))+1;
us:=trunc(ln(w)/ln(10));
outTextxy(437,235,'w(rad/s)');

```



```

outTextxy(52,20,chr(179)+'A');
outTextxy(70,23,'vs');
outTextxy(85,20,chr(179));
outTextxy(92,25,'dB');
outTextxy(94,alm-3,'');
outTextxy(119,alm-3,'');
outTextxy(138,alm-3,'');
outTextxy(153,alm-3,'');
outTextxy(165,alm-3,'');
outTextxy(175,alm-3,'');
outTextxy(183,alm-3,'');
outTextxy(191,alm-3,'');
outTextxy(243,alm-3,'');
outTextxy(269,alm-3,'');
outTextxy(288,alm-3,'');
outTextxy(303,alm-3,'');
outTextxy(315,alm-3,'');
outTextxy(325,alm-3,'');
outTextxy(333,alm-3,'');
outTextxy(341,alm-3,'');
outTextxy(393,alm-3,'');
outTextxy(419,alm-3,'');
outTextxy(438,alm-3,'');
outTextxy(453,alm-3,'');
outTextxy(465,alm-3,'');
outTextxy(475,alm-3,'');
outTextxy(483,alm-3,'');
outTextxy(491,alm-3,'');
aps(48,alm+7,63,alm+4,us-1);
outTextxy(93,alm+7,'2');
outTextxy(137,alm+7,'4');
outTextxy(165,alm+7,'6');
outTextxy(183,alm+7,'8');
aps(195,alm+7,210,alm+4,us);
aps(345,alm+7,360,alm+4,us+1);
aps(495,alm+7,510,alm+4,us+2);
outTextxy(243,alm+7,'2');
outTextxy(288,alm+7,'4');
outTextxy(315,alm+7,'6');
outTextxy(333,alm+7,'8');
outTextxy(393,alm+7,'2');
outTextxy(437,alm+7,'4');
outTextxy(465,alm+7,'6');
outTextxy(483,alm+7,'8');
if avs>1 then
outTextxy(40,ulm-5+trunc(6*(alm-ulm)/9),'0')
else
outTextxy(40,ulm-5,'0');
line(50,ulm+trunc(6*(alm-ulm)/9),
500,ulm+trunc(6*(alm-ulm)/9));
line(50,20,50,alm);
line(200,alm,200,ulm);
line(350,alm,350,ulm);
line(500,alm,500,ulm);

```

```

for i:=0 to 9 do
begin
line(50,ulm+trunc(i*(alm-ulm)/9),
      500,ulm+trunc(i*(alm-ulm)/9));
end;
if avs>1 then begin
for i:=1 to 6 do
Begin
str(i*bry,sayi);
if i*bry>9 then ap:=32 else ap:=40;
outTextxy(ap,ulm+trunc(6*br-5-i*br),sayi);
end;
for i:=1 to 3 do
Begin
str(-i*bry,sayi);
if i*bry>9 then ap:=24 else ap:=32;
outTextxy(ap,ulm+trunc(6*br-5+i*br),sayi);
end;
end;
if avs<=1 then
for i:=1 to 9 do
Begin
str(-i*bry,sayi);
if i*bry>9 then ap:=24 else ap:=32;
outTextxy(ap,ulm-5+trunc(i*br),sayi);
end;

putpixel(49,21,white);
putpixel(48,22,white);
putpixel(51,21,white);
putpixel(52,22,white);
outTextxy(510,300,'w');
if f=2 then outTextxy(518,303,'L')
           else outTextxy(518,303,'H');
outTextxy(526,300,'=');
if w<1E+06 then
begin
str(trunc(w),st);
str(trunc(10*frac(w)),say);
str(trunc(100*frac(w))-10*trunc(10*frac(w)),
    sayi);
outTextxy(534,300,st+ '.'+say+sayi+'rad/s') ;
end
else
begin
str(trunc(w/1E+06),st);
str(trunc(10*frac(w/1E+06)),say);
str(trunc(100*frac(w/1E+06))-
    10*trunc(10*frac(w/1E+06)),sayi);
outTextxy(534,300,st+ '.'+say+sayi+'Mrad/s' );
end;
if avs>1 then begin
i:=50;j:=0;
repeat

```

```

putpixel(i,50+trunc(6*br)-
trunc((br/bry)*20*ln(Avs/kok)/ln(10)),white);
j:=j+1;if j=6 then begin i:=i+4;j:=0;end
      else i:=i+1;
until i>50+trunc(150*ln(w)/ln(10))-(us-1)*150;
i:=50+trunc(6*br)-trunc((br/bry)*20*ln(0.707*Avs)/
ln(10));j:=0;
repeat
putpixel(trunc(150*ln(w)/ln(10))-(us-1)*
      150+50,i,white);j:=j+1;
if j=6 then begin i:=i+4;j:=0;end else i:=i+1;
until i>50+trunc(6*br);
outTextxy(trunc((ln(w)/ln(10))*150-(us-1)*
150+35),34+trunc(6*br),'w');
if f=2 then
outTextxy(trunc((ln(w)/ln(10))*150-(us-1)*
      150+43),40+trunc(6*br),'L')
      else
outTextxy(trunc((ln(w)/ln(10))*150-(us-1)*
150+43),40+trunc(6*br),'H');
aps(52,60+trunc(6*br),67,57+trunc(6*br),us-1);
aps(202,60+trunc(6*br),217,57+trunc(6*br),us);
aps(352,60+trunc(6*br),367,57+trunc(6*br),us+1);
aps(475,60+trunc(6*br),490,57+trunc(6*br),us+2);
      end;
if avs<=1 then begin
i:=50;j:=0;
repeat
putpixel(i,ulm-trunc((br/bry)*20*ln(Avs/kok)/
ln(10)),white);j:=j+1;
if j>6 then begin i:=i+4;j:=0;end
      else i:=i+1;
until i>50+trunc(150*ln(w)/ln(10))-(us-1)*150;
i:=50;j:=0;
repeat
putpixel(trunc(150*ln(w)/ln(10))-(us-
1)*150+50,i,white);j:=j+1;
if j=6 then begin i:=i+4;j:=0;end
      else i:=i+1;
until i>=ulm-
trunc((br/bry)*20*ln(Avs/kok)/ln(10));
outTextxy(trunc((ln(w)/ln(10))*
      150-(us-1)*150+35),51,'w');
if f=2 then
outTextxy(trunc((ln(w)/ln(10))*
      150-(us-1)*150+43),57,'L')
      else
outTextxy(trunc((ln(w)/ln(10))*
      150-(us-1)*150+43),57,'H');
aps(52,40,67,33,us-1);
aps(202,40,217,33,us);
aps(352,40,367,33,us+1);
aps(475,40,490,33,us+2);
      end;

```

```

w:=exp((us-1)*ln(10));wx:=exp((us+2)*ln(10));
repeat
wd:=w/100;
w:=w+wd;
xa:=trunc(150*ln(w)/ln(10));
xb:=trunc((br/bry)*20*ln(Avs/fonk(w))/ln(10));
if avs>1 then if 50+trunc(6*br)-xb<alm then
putpixel(xa-(us-1)*150+50,50+
trunc(6*br)-xb,white);
if avs<=1 then if ulm-xb<alm then
putpixel(xa-(us-1)*150+50,ulm-xb,white);
until w>=wx;
OutTextxy(10,450,'< ESC > Menu');
Repeat ch:=ReadKey;until ch=chr(27);
CloseGraph;
end
else
WriteLn('Graphics error:',
GraphErrorMsg(ErrCode));
END;
PROCEDURE chesabi;
var cp1,cp2,bufer:real;
BEGIN
if e=1 then rb:=paralel(r11,r21)
else rb:=zi;
case e of
1:begin
case b of
1:begin
c1:=2E-02/(rs+paralel(rp,rb))/wl;
c2:=2E-02/(rl+rc)/wl;
ce:=1E-03/paralel((rp+paralel(rs,rb))/
(b0+1),re)/ wl;
end;
2:begin
c1:=1E-03/(rs+zi)/wl;
c2:=2E-02/(rc+r1)/wl;
rth:=paralel(rb,(rp+(b0+1)*
paralel(rs,re)));
c3:=2E-02/rth/wl;
end;
3:begin
zi:=paralel(rb,(rp+(b0+1)*paralel(re,r1)));
z0:=paralel(re,(rp+paralel(rb,rs))/(b0+1));
c1:=2E-02/(rs+zi)/wl;
c2:=1E-03/(rl+z0)/wl;
end;
end;
f:=2;
repeat
whesabi;
if w>wl then case b of
1:ce:=1.01*ce;
2:c1:=1.01*c1;

```

```

3:c2:=1.01*c2;
end;
until (w<=w1) or (w<0);
cmu0:=cmu*sqrt(1+vcb/0.8);
vcb:=vce-v0;cmu:=cmu0/sqrt(1+vcb/0.8);
f:=3;
cp1:=0;cp2:=1;cp:=1;
repeat
if fonk(wh)>kok then cp2:=(cp1+cp2)/2
else cp1:=(cp1+cp2)/2;
bufer:=fonk(wh);
cp:=(cp1+cp2)/2;
until (fonk(wh)-kok<=1E-03) and (fonk(wh)-kok>0)
or (abs(fonk(wh)-bufer)<=1E-03);
end;
2,3:begin
case b of
1:begin
c1:=2E-02/(rs+rb)/w1;
c2:=2E-02/(rd+rl)/w1;
if rss<>0 then
css:=1E-03/ paralel(rss,(1/gm))/w1;
end;
2:begin
c1:=1E-03/(rs+zi)/w1;
c2:=2E-02/(rd+rl)/w1;
if rb<>0 then c3:=2E-02/rb/w1;
end;
3:begin
c1:=2E-02/(rs+rb)/w1;
c2:=1E-03/(rl+z0)/w1;
end;
end;
f:=2;
repeat
whesabi;
if w>w1 then case b of
1:css:=1.01*css;
2:c1:=1.01*c1;
3:c2:=1.01*c2;
end;
until (w<=w1) or (w<0);
f:=3;
cp1:=0;cp2:=1;cgs:=1;
repeat
if fonk(wh)>kok then cp2:=(cp1+cp2)/2
else cp1:=(cp1+cp2)/2;
bufer:=fonk(wh);
cgs:=(cp1+cp2)/2;
until (fonk(wh)-kok<=1E-03) and (fonk(wh)-kok>0)
or (cgs=0) or (abs(fonk(wh)-bufer)<=1E-03);
end;
end;
if c1>=1E-09 then writeln('C1 =',c1*1E+06:4:3,' pF')

```

```

else writeln('C1 =',c1*1E+12:4:3,' ','pF');
if c2>=1E-09 then writeln('C2 =',c2*1E+06:4:3,' μF')
else writeln('C2 =',c2*1E+12:4:3,' ','pF');
if e=1 then
begin
if b=1 then if ce>=1E-09 then
writeln('CE =',ce*1E+06:4:3,' μF')
else
writeln('CE =',ce*1E+12:4:3,' ','pF');
if b=2 then if c3 >=1E-09 then
writeln('C3 =',c3*1E+06:4:3,' μF')
else
writeln('C3 =',c3*1E+12:4:3,' ','pF');
end;
if e<>1 then
begin
if b=1 then
if rss<>0 then if css>=1E-09 then
writeln('Css=',css*1E+06:4:2,' μF')
else
writeln('Css=',css*1E+12:4:2,' pF')
else begin writeln('Css=∞');css:=-1;end;
if b=2 then
if c3>=1E-09 then writeln('C3=',c3*1E+06:4:2,' μF')
else writeln('C3=',css*1E+12:4:2,' pF');
end;
if e=1 then if (abs(fonk(wh)-kok)<=1E-03) then
begin
writeln('Cp =',cmu*1E+12:3:2,' pF');
writeln('Cp =',cp*1E+12:3:2,' pF');
writeln('fT =',gm*1E-09/(cp+cmu)/2/pi:4:2,' MHz');
end
else
begin
whesabi;
writeln('WH < ',w/1E+06:4:2,' Mrad/s seçiniz');
end
else if (abs(fonk(wh)-kok)<=1E-03) then
begin
writeln('Cgd =',cgd*1E+12:3:2,' pF');
writeln('Cgs =',cgs*1E+12:3:2,' pF');
end
else
begin
whesabi;
writeln('WH < ',w/1E+06:4:2,' Mrad/s
seçiniz');
end;
end;
END;
PROCEDURE tbipolar;
BEGIN
window(3,3,77,23);
if not ((ord(ch1)=27) and (ord(ch2)=0)) then
begin

```

```

avs:=abs(avs);bdc:=bdcmin;b0:=b0min;
z:=zi/(rs+zi);
if b=1 then begin
    rc:=z0;gm:=avs/paralel(rc,r1)/z;
    end;
if b=2 then begin rc:=z0;gm:=avs/paralel(rc,r1)/z;end;
if b=3 then gm:=38.92;
rp:=b0/gm;
ic:=gm/38.92;
zi0:=zi;avs0:=avs;z00:=z0;f0:=f;f:=0;kk:=50;
repeat
gm:=38.92*ic;rp:=b0/gm;
if b=1 then if rd<>-1 then
rc:=paralel(r1,rd)*avs0/(z*gm*paralel(r1,rd)-avs0)
    else rc:=r1*avs0/(z*gm*r1-avs0);
if b<>3 then vre:=(vcc-rc*ic)/2 else vre:=vcc/3;
re:=vre/ic;ib:=ic/bdc;
r11:=(vcc-vre-v0)/kk/ib;r21:=(vre+v0)/(kk-1)/ib;
dcbipolar;
case b of
1:begin
    offbipemt;
    if zi<zi0 then
    if rp<zi0 then ic:=0.99*ic
        else kk:=kk-1
        else if abs(avs)<avs0 then rc:=rc+0.1;
    end;
2:begin
    offbipbaz;
    if zi>zi0 then ic:=1.01*ic
        else if avs<avs0 then rc:=rc+0.1;
    end;
3:begin
    offbipcol;
    if zi<zi0 then kk:=kk-1
        else if z0>z00 then re:=re-0.1;
    end;
end;
until ((b=1) and (zi>=zi0) and (abs(avs)>=avs0)) or
((b=2) and (zi<=zi0) and (abs(avs)>=avs0)) or
((b=3) and (zi>zi0) and (z0<=z00)) or (kk<1) or
(re<0);
window(43,3,77,23);
if (re<0) or (kk<1) or (vce<0) then
begin
window(10,23,40,23);textcolor(15+blink);
write('AMPLi TASARIMI YAPILAMADI');textcolor(7);
end
    else begin
clrscr;
writeln('Vcc=',vcc:2:0,' V');
writeln('R1 =',r11:4:2,' kΩ');
writeln('R2 =',r21:4:2,' kΩ');
if b<>3 then writeln('RC ≥',rc:4:2,' kΩ');

```

```

writeln('RE =',re:4:2,' kΩ');
bdc:=bdc; b0:=b0m; dcbipolar;
case b of
  1: ofbipemt;
  2: ofbipbaz;
  3: ofbipcol;
end;
chesabi;
end;
end;
f:=f0;
END;
PROCEDURE tfet;
var vg,p,q,r:real;
BEGIN
f0:=f;
z:=zi/(rs+zi); if b=3 then rd:=0 else rd:=z0;
case e of
  2: case b of
    1: begin
      gm:=avs/paralel(rd,rl)/z;
      vgs:=gm*sqr(vp)/2/idss+vp;
      id:=idss*sqr(vgs-vp)/sqr(vp);
      rss:=-vgs/id;
      r2l:=zi; rk:=1;
      end;
    2: begin
      gm:=avs/paralel(rd,rl)/z;
      vgs:=gm*sqr(vp)/2/idss+vp;
      id:=idss*sqr(vgs-vp)/sqr(vp);
      rss:=zi/(1-gm*zi);
      zi0:=zi; avs0:=avs;
      repeat
        f:=0; rk:=1; dcfet; offetbaz;
        if (zi>zi0) or (avs<avs0) and
          (vds<=vgs-vp) then rss:=rss-0.01;
        if (avs<avs0) and (zi<=zi0) and
          (vds>vgs-vp) then rd:=rd+0.1;
        until (avs>=avs0) and (zi<=zi0) or
          (rss<0);
      end;
    3: begin
      p:=vp*vp/4/idss/z0/z0;
      q:=vp; r:=-vp*vp/4/idss;
      rss:=(-q-sqrt(q*q-4*p*r))/2/p;
      if (rss<0) then
        rss:=(-q+sqrt(q*q-4*p*r))/2/p;
      r2l:=zi; rk:=1;
      end;
  end;
3: case b of
  1: begin
    gm:=avs/paralel(rd,rl)/z;
    vgs:=(gm+kk1*vt)/kk1;

```



```

        id:=kk1*sqr(vgs-vt)/2;
        vg:=vgs+(vdd-rd*id)/2;
        r11:=zi*vdd/vg;r21:=zi*r11/(r11-zi);
        rss:=(vdd-rd*id)/2/id;
        end;
    2:begin
        gm:=avs/paralel(rd,r1)/z;
        vgs:=(gm+kk1*vt)/kk1;
        id:=kk1*sqr(vgs-vt)/2;
        rss:=zi/(1-gm*zi);
        vg:=vgs+rss*id;
        r11:=1000;
        r21:=r11*vg/(vdd-vg);
        end;
    3:begin
        p:=1/2/kk1/z0/z0;q:=vt-vdd;r:=-1/2/kk1;
        rss:=(-q-sqrt(q*q-4*p*r))/2/p;
        if (rss<0) then rss:=(-q+sqrt(q*q-
        4*p*r))/2/p;
        r11:=zi;rk:=1;
        end;
    end;
end;
if rss<0 then
begin
window(10,16,40,16);textcolor(15+blink);
write('AMPLi TASARIMI YAPILAMADI');textcolor(7);
end
else begin
if e=2 then dcfet else dcarttirilmis;
window(50,3,77,23);
clrscr;
if e=3 then begin writeln('R1 =',r11:4:2,' kΩ');
                    if r11=zi then begin writeln('R2
=∞');r21:=0;end
                                else writeln('R2
=',r21:4:2,' kΩ');
                    end
                else
begin
r11:=0;
writeln('R1 = ∞');
if b<>2 then begin writeln('R2 =',zi:4:2,'
kΩ');r21:=zi;end
                else writeln('R2 =0');
end;
if b<>3 then writeln('RD ≥ ',rd:4:2,' kΩ');
if rss>=1 then writeln('Rss=',rss:5:3,' kΩ')
                else writeln('Rss=',rss*1000:1:1,' Ω');
chesabi;
end;
f:=f0;
END;
PROCEDURE sifirla;

```

```

BEGIN
vcc:=0;avs:=0;rs:=0;rl:=0;zi:=0;z0:=0;b0:=0;b0m:=0;
bdc:= 0;bdcmin:=0;vcb:=0;bdcmin:=0;b0min:=0;cp:=0;cmu:=0;
wl:=0;wh:=0;vp:=0;idss:= 0;vdd:=0;r1l:=0;r2l:=0;rc:=0;
re:=0; rss:=0;c1:=0;c2:=0;c3:=0;ce:=0;css:=0;cgd:=0;
cgs:=0; v0:=0.65;R0:=-1;rd:=-1;RBB:=0;ICE0:=0;rx:=0;
rk:=0;vt:=0;kk1:=0;
END;
PROCEDURE kaydet;
BEGIN
if (fileexists(da)) and (d=2) then
begin
write(#7);
pencere(23,11,59,16);
gotoxy(2,1);write(' ','da,' dosyasi mevcut !');
gotoxy(2,2);write('üzerine kaydetmek istiyormusunuz?');
gotoxy(2,4);write('          ( E/H )');
imlec;
repeat
chl:=readkey;
until upcase(chl) in ['E','H'];
if upcase(chl)='E' then begin
str1:=copy(da,1,x1)+'bak';
if fileexists(str1) then begin
assign(dos,str1);erase(dos);end;
assign(dos,da);
rename(dos,str1);kayit:=1;
end
else exit;
end;
assign(dos,da);
ReWrite(dos);
if (e=1) and (k=1) then
Write(dos,kebl,vcc,r1l,r2l,rc,re,rs,rl,b0,bdc,c1,c2,c3,
c e,cp,cmu,rbb,r0,rd,rx,v0,ice0);
if (e<>1) and (k=1) then
Write(dos,kebl,vdd,r1l,r2l,rd,rss,rs,rl,vp,idss,vt,kk1,
c 1,c2,c3,css,cgd,cgs);
if (e=1) and (k=2) then
Write(dos,kebl,vcc,avs,rs,rl,zi,z0,b0min,b0m,bdcmin,bdcmin,
vcb,cmu,wl,wh,rbb,rd,rx,v0,ice0,r0);
if (e<>1) and (k=2) then
Write(dos,kebl,vdd,avs,zi,z0,wl,wh,cgd,rs,rl,vp,idss,vt,
kk1);
close(dos);kayit:=1;
END;
PROCEDURE kayitsor;
BEGIN
menu(i1);write(#7);
pencere(16,11,63,16);
gotoxy(2,1);write(' ','da,' dosyasi degisti');
gotoxy(2,2);write('Dosyada yapılan degisiklikler
saklansın mı ?');
gotoxy(2,4);write('          ( E/H )');

```

```

imlec;
repeat
ch:=readkey;
until upcase(ch) in ['E','H'];
if upcase(ch)='E' then kaydet else iptal:=1;
END;
PROCEDURE daoku;
label kont;
BEGIN
if (d=1) and (kayit=0) and (ilkchar=1) and (da<>'') and
(iptal=0) then kayitsor;
window(3,25,77,25);textcolor(0);textbackground(7);
gotoxy (2,1);write('< ESC > MENU');
window(3,3,77,23);textcolor(7);textbackground(0);clrscr;
getdir(0,str1);da:='';
write('Dosya adi:',str1);if
copy(str1,length(str1),1)<>'\' then write('\');
if dabuf<>'\' then begin
write(dabuf);
gotoxy(wherex-length(dabuf),wherey);
ch2:=#0;
ch1:=readkey;if keypressed then ch2:=readkey;
if (ch1=#13) and (ch2=#0) then begin da:=dabuf;goto
kont;end
else
if (ch1=#27) and (ch2=#0) then exit
else
if (ord(ch1) in [46,48..57,65..90,97..122]) and (ch2=#0)
then
begin
write(' ':length(dabuf));
gotoxy(wherex-length(dabuf),wherey);
ch1:=upcase(ch1);da:=ch1; write(ch1);
end;
end;
repeat
ch2:=#0;
ch1:=readkey;if keypressed then ch2:=readkey;
if (ord(ch1) in [8,13,27,46,48..57,65..90,97..122]) and
(ord(ch2)=0) then
begin
if (ord(ch1)=27) and (ord(ch2)=0) then exit;
if (ord(ch1)=8) and (ord(ch2)=0) and
(wherex>12+length(str1)) then begin

da:=copy(da,1,length(da)-1);
gotoxy(wherex-1,wherey);write(' ');
gotoxy(wherex-1,wherey);
end;
if (ord(ch1) in [46,48..57,65..90,97..122]) and (ch2=#0)
then
begin ch1:=upcase(ch1);write(ch1);da:=da+ch1;end;
end else write(#7);
until (ch1=#13) and (ch2=#0) and (da<>'');

```

```

kont:
i:=0;x2:=0;x1:=0;
repeat
i:=i+1;
if copy(da,i,1)='.' then begin x2:=x2+1;x1:=i;end;
until i=12;
if (copy(da,1,1)='.') or (x2>1) then
begin
clrscr;write(#7);
pencere(27,11,52,13);
gotoxy(5,2);write('Dosya ismi hatalı !');
imlec;
repeat
chl:=readkey;if keypressed then chl:=readkey;
until (chl=#27) and (ch2=#0);
exit;
end;
if x1<>0 then str1:=copy(da,1,x1)+copy(da,x1+1,3)
else str1:=copy(da,1,8);
da:=str1;dabuf:=da;if d=1 then sifirla;
END;
PROCEDURE degoku;
BEGIN
assign(dos,da);
reset(dos);
read(dos,keb2);
if kebl<>keb2 then
begin
write(#7);
pencere(20,11,54,14);
write(' ',da,' dosyasının içeriği');
gotoxy(2,2);write(' seçeneklere uygun değil !');
imlec;
repeat
chl:=readkey;if keypressed then chl:=readkey;
until (chl=#27) and (ch2=#0);
exit;
end;
if (e=1) and (k=1) then
Read(dos,vcc,r11,r21,rc,re,rs,r1,b0,bdc,c1,c2,c3,ce,cp,
c mu,rbb,r0,rd,rx,v0,ice0);
if (e<>1) and (k=1) then
begin
Read(dos,vdd,r11,r21,rd,rss,rs,r1,vp,idss,vt,kk1,c1,c2,
c3,css,cgd,cgs);
if e=2 then if r11=0 then rk:=1 else rk:=0;
if e=3 then if r21=0 then rk:=1 else rk:=0;
end;
if (e=1) and (k=2) then
Read(dos,vcc,avs,rs,r1,zi,z0,b0min,b0m,bdcmin,bdcm,vcb,
cmu,wl,wh,rbb,rd,rx,v0,ice0,r0);
if (e<>1) and (k=2) then
Read(dos,vdd,avs,zi,z0,wl,wh,cgd,rs,r1,vp,idss,vt,kk1);
close(dos);dosk:=1;

```

```

END;
PROCEDURE tuskont;
BEGIN
  repeat
    chl:=readkey;if keypressed then chl:=readkey;
  until (chl=#27) and (ch2=#0);
  textbackground(0);textcolor(7);
END;
PROCEDURE dosmenu;
BEGIN
  menu(i1);
  kebl:=k*100+e*10+b;
  if d=1 then begin
    daoku;
    if ((chl=#27) and (ch2=#0)) then exit;
    if fileexists(da) then begin degoku;if (chl=#27) and
      (ch2=#0) then exit;end;
    textcolor(0);textbackground(7);window(1,25,79,25);
    gotoxy(67,1);write(da);textcolor(7);textbackground(0);
    window(3,3,77,23);
    editor(vcc,vdd,r1l,r2l,rc,rd,r1,re,rss,rs,bdc,b0,vp,vt,
      i dss,kk1,c1,c2,c3,ce,css,cp,cmu,cgs,cgd,ice0,
      v0,rx,r0,avs,zi, wl,wh,rbb,vcb,z0,b0min,b0m,
      bdcmin, bdcmin,k,k1,e,b,rk,ins,ilkc har,chl,ch2);
    if (chl=#27) and (ch2=#0) and (ilkchar=0) then
      begin sifirla;dosk:=0;iptal:=1;end
      else
      begin
        dosk:=1;kayit:=0;
        if bdc=0 then bdc:=bdcmin;
        if b0=0 then b0:=b0min;
        if b0min=0 then b0min:=b0;
        if bdcmin=0 then bdcmin:=bdc;
      end;
    if d=2 then if (dosk=0) and (ilkchar=0) then
      begin
        write(#7);
        pencere(27,11,49,13);
        write(' Açılmış dosya yok !');
        window(3,25,77,25);gotoxy(2,1);write('< ESC > MENU');
        imlec;
        repeat
          chl:=readkey;if keypressed then chl:=readkey;
        until (chl=#27) and (ch2=#0);
        exit;
      end
      else begin
        daoku;
        if not ((chl=#27) and
          (ch2=#0)) then kaydet;exit;
        end;
    if d=3 then begin daoku;
      if fileexists(da) then begin
        erase(dos);exit;end

```

```

else
begin
write(#7);
pencere(25,10,53,12);
gotoxy(2,2);write(' ',da,' bulunamadi !');
window(3,25,77,25);
gotoxy(2,1);write('< ESC > MENU');
imlec;
tuskont;exit;
end;
end;
if d=4 then begin
daoku;assign(dos,da);
if fileexists(da) then begin
reset(dos);
{$I-}
read(dos,keb2);
{$I+}
if IOResult<>0 then
begin
pencere(32,10,50,15);
write('Hatali dosya !');
close(dos);imlec;tuskont;
end;
x2:=trunc(keb2);
pencere(3,10,78,15);
str1:='Ortak ';
k1:=x2 -trunc(x2/10)*10;
case k1 of
1:str1:=str1+'emetör ';
2:str1:=str1+'baz ';
3:str1:=str1+'kollektör ';
else begin
pencere(32,10,50,12);
write(#7);write(' Hatali dosya !');imlec;
tuskont;exit;
end;
end;
str1:=str1+'baglantili ';
k1:=trunc(frac(x2/100)*10);
case k1 of
1:str1:=str1+'bipolar transistörlü ';
2:str1:=str1+'jfet veya mosfetli ';
3:str1:=str1+'arttirilmis mosfetli ';
else
begin
pencere(32,10,50,12);write(#7);
write(' Hatali dosya !');imlec;
tuskont;exit;
end;
end;
str1:=str1+'amplifikatörün ';
i:=length(str1);
k1:=trunc(x2/100);

```

```

case k1 of
  1:str1:=str1+'analizi ';
  2:str1:=str1+'tasarımı ';
  else
begin pencere(32,10,50,12);write(#7);
  write(' Hatalı dosya !');imlec;
  tuskont;exit;
end;
end;
str1:=str1+'ile ilgili veri dosyası';
gotoxy(trunc((75-i)/2),2);write(copy(str1,1,i));
gotoxy(trunc((75-(length(str1)-i))/2),3);
write(copy(str1,i,length(str1)-i+1));
imlec;
tuskont;
                                end else begin
write(#7);pencere(25,10,53,12);
write(' ',da,' bulunamadı !');
imlec;tuskont;
                                end;
                                end;
END;
Begin
kok:=exp(0.5*ln(2));da:='';
sifirla;menu(i1);TextBackGround(0);TextColor(7);
d:=1;k:=1;f:=1;e:=1;b:=1;c:=1;
oku:
j2:=0;ch1:=#0;ch2:=#0;
ch1:=readkey;if keypressed then ch2:=readkey;
j1:=Ord(ch1);j2:=ord(ch2);j3:=1;
  Case j1 of
    27:if j2=0 then
Begin dm:=0;km:=0;fm:=0;em:=0;bm:=0;menu(i1);end;
    0:Case j2 of
      32:begin
        i1:=1;
        dosya(dm,r1,r2,r3,r4,r5,r6,r7,r8);
        i1:=1;
        end;
      37:begin
        i1:=2;konu(km,r1,r2,r3,r4);
        i1:=2;end;
      33:begin
        i1:=3;
        frekans(fm,r1,r2,r3,r4,r5,r6);
        i1:=3;
        end;
      18:begin
        i1:=4;
        eleman(em,r1,r2,r3,r4,r5,r6);
        i1:=4;end;
      48:begin
        i1:=5;
        baglanti(bm,r1,r2,r3,r4,r5,r6);

```

```

        il:=5;end;
19:begin il:=6;run;end;
46:begin il:=7;cikis;end;
77:begin
    il:=il+1;
    if (k=2) and (il=3) then il:=4;
    end;
75:begin
    il:=il-1;
    if (k=2) and (il=3) then il:=2;
    end;
72:case il of
    1:if dm=1 then d:=d-1;
    2:if km=1 then k:=k-1;
    3:if fm=1 then f:=f-1;
    4:if em=1 then e:=e-1;
    5:if bm=1 then b:=b-1;
    end;
80:case il of
    1:if dm=1 then d:=d+1;
    2:if km=1 then k:=k+1;
    3:if fm=1 then f:=f+1;
    4:if em=1 then e:=e+1;
    5:if bm=1 then b:=b+1;
    end;
end;
73,105 :if (dm=1) and (il=1) then d:=3;
75,107 :if (bm=1) and (il=5) then b:=3;
69,101 :if (bm=1) and (il=5) then b:=1;
66,98  :begin
    if (dm=1) and (il=1) then d:=4;
    if (em=1) and (il=4) then e:=1;
    if (bm=1) and (il=5) then b:=2;
    End;
89,121 :if (fm=1) and (il=3) then f:=3;
65,97  :begin
    if (dm=1) and (il=1) then d:=1;
    if (km=1) and (il=2) then k:=1;
    if (fm=1) and (il=3) then f:=2;
    if (em=1) and (il=4) then e:=3;
    end;
83,115 :if (dm=1) and (il=1) then d:=2;
84,116 :begin
    if (km=1) and (il=2) then k:=2;
    if (fm=1) and (il=3) then f:=1;
    if (em=1) and (il=4) then e:=2;
    end;
79,111 :if (fm=1) and (il=3) then f:=1;
74,106 :if (em=1) and (il=4) then e:=2;
13 :Begin
    keb:=k*100+e*10+b;
    if (il=1) and (dm=1) then dosmenu;
    dm:=1;km:=1;fm:=1;em:=1;bm:=1;
    case il of

```



```

6:if dosk=1 then begin

if keb<>kebl then
begin
menu(il);
write(#7);
pencere(20,11,54,14);
        write(' ',da,' dosyasının içeriği');
gotoxy(2,2);write(' seçeneklere uygun değil !');
imlec;tuskont;
clrscr;
goto oku;
end else
if (f=1) or (k=2) then begin
if (ilkchar=1) and (kayit=0) and (da<>'') then kayitsor;
if fileexists(da) then degoku;
textcolor(7);textbackground(0);window(3,3,79,23);clrscr;
textcolor(7);textbackground(0);window(3,3,41,23);
if (e=1) and (k=2) then
tablo1(vcc,avs,rs,rl,zi,z0,b0min,b0m,bdcmin,bdcm,vcb,
        cmu,wl,wh,rbb,rd,rx,v0,ice0,r0,b);
if (e=1) and (k=1) then
tablo2(vcc,r1l,r2l,rc,re,rs,rl,b0,bdc,c1,c2,c3,ce,cp,
        cmu ,rbb,r0,rd,rx,v0,ice0,b);
if (e<>1) and (k=1) then
tablo3(vdd,r1l,r2l,rd,rss,rs,rl,vp,idss,vt,kk1,c1,c2,c3,
        css,cgd,cgs,e,b,rk);
if (e<>1) and (k=2) then
tablo4(vdd,avs,zi,z0,wl,wh,cgd,rs,rl,vp,idss,vt,kk1,
        e,b) ;
                                end;
                                if (ord(ch1)=27) and
                                (ord(ch2)=0) then menu(il) else

case k of
1:case f of
1:case e of
1:begin dcbipolar;
case b of
1:ofbipemt;
2:ofbipbaz;
3:ofbipcol;
end;
end;
2,3:begin if e=2 then dcfet
else dcarttirilmis;
case b of
1:offetemt;
2:offetbaz;
3:offetcol;
end;
End;
End;
2,3:begin
case e of

```

```

1:begin dcbipolar;
  case b of
    1:ofbipemt;
    2:ofbipbaz;
    3:ofbipcol;
  end;
end;
2,3:begin
  if e=2 then dcfet
    else dcarttirilmis;
  case b of
    1:offetemt;
    2:offetbaz;
    3:offetcol;
  end;
end;
end;
  afbip;menu (il);
end;
End;
2:begin
  case e of
    1:tbipolar;
    2,3:tfet;
  end;
end;
End;
if f=1 then begin
window(3,25,79,25);textcolor(0);textbackground(7);
gotoxy(2,1);
write('< ESC > MENU ');gotoxy(65,1);write(da);
imlec;tuskont;menu(il);
end;
goto oku;
End
else begin
write(#7);
pencere(27,11,49,13);
gotoxy(2,2);write(' Açılmış dosya yok !');
window(3,25,77,25);gotoxy(2,1);write('< ESC > MENU');
imlec;tuskont;
end;
1:dosya(dm,r1,r2,r3,r4,r5,r6,r7,r8);
2:konu(km,r1,r2,r3,r4);
3:frekans(fm,r1,r2,r3,r4,r5,r6);
4:eleman(em,r1,r2,r3,r4,r5,r6);
5:baglanti(bm,r1,r2,r3,r4,r5,r6);
7:if (kayit=1) or (ilkchar=0) or
    (iptal=1) then goto son
else
begin
if (da<>'') and (kayit=0) and (ilkchar=1) then
kayitsor;goto son;

```

```

end;

                                end;
                                End;

End;
If i1<1 Then i1:=7;
If i1>7 Then i1:=1;
    Case i1 Of
        1:Begin if d>4 then d:=1;
                    if d<1 then d:=4;
                case d of
                    1:Begin
                        r1:=0;r2:=7;r3:=7;r4:=0;r5:=7;r6:=0;
                        r7:=7 ;r8:=0;menu(i1);
                        dosya(dm,r1,r2,r3,r4,r5,r6,r7,r8);
                        End;
                    2:Begin
                        r1:=7;r2:=0;r3:=0;r4:=7;r5:=7;r6:=0;
                        r7:=7 ;r8:=0;menu(i1);
                        dosya(dm,r1,r2,r3,r4,r5,r6,r7,r8);
                        End;
                    3:Begin
                        r1:=7;r2:=0;r3:=7;r4:=0;r5:=0;r6:=7;
                        r7:=7 ;r8:=0;menu(i1);
                        dosya(dm,r1,r2,r3,r4,r5,r6,r7,r8);
                        End;
                    4:Begin
                        r1:=7;r2:=0;r3:=7;r4:=0;r5:=7;r6:=0;
                        r7:=0 ;r8:=7;menu(i1);
                        dosya(dm,r1,r2,r3,r4,r5,r6,r7,r8);
                        End;
                end;
            end;
        2:Begin If k>2 Then k:=1;
                    If k<1 Then k:=2;
                    Case k Of
                        1:Begin
                            r1:=0;r2:=7;r3:=7;r4:=0;
                            menu(i1);konu(km,r1,r2,r3,r4);
                            End;
                        2:Begin
                            r1:=7;r2:=0;r3:=0;r4:=7;
                            menu(i1);konu(km,r1,r2,r3,r4);
                            End;
                    End;
                End;
            End;
        3:Begin If f>3 Then f:=1;
                    If f<1 Then f:=3;
                    Case f Of
                        1:Begin
                            r1:=0;r2:=7;r3:=7;r4:=0;
                            r5:=7;r6:=0;menu(i1);
                            frekans(fm,r1,r2,r3,r4,
                                r5,r6);
                            End;

```

```

2:Begin
  r1:=7;r2:=0;r3:=0;r4:=7;
  r5:=7;r6:=0;menu(i1);
  frekans(fm,r1,r2,r3,r4,
           r5,r6);
  End;
3:Begin
  r1:=7;r2:=0;r3:=7;r4:=0;
  r5:=0;r6:=7;menu(i1);
  frekans(fm,r1,r2,r3,r4,
           r5,r6);
  End;
End;
End;
4:Begin If e>3 Then e:=1;
        If e<1 Then e:=3;
        Case e Of
          1:Begin
            r1:=0;r2:=7;r3:=7;r4:=0;
            r5:=7;r6:=0;menu(i1);
            eleman(em,r1,r2,r3,r4,r5,r6);
            End;
          2:Begin
            r1:=7;r2:=0;r3:=0;r4:=7;
            r5:=7;r6:=0;menu(i1);
            eleman(em,r1,r2,r3,r4,r5,r6);
            End;
          3:Begin
            r1:=7;r2:=0;r3:=7;r4:=0;
            r5:=0;r6:=7;menu(i1);
            eleman(em,r1,r2,r3,r4,r5,r6);
            End;
        End;
End;
End;
5:Begin If b>3 Then b:=1;
        If b<1 Then b:=3;
        Case b Of
          1:Begin
            r1:=0;r2:=7;r3:=7;r4:=0;
            r5:=7;r6:=0;menu(i1);
            baglanti(bm,r1,r2,r3,r4,
                     r5,r6);
            End;
          2:Begin
            r1:=7;r2:=0;r3:=0;r4:=7;
            r5:=7;r6:=0;menu(i1);
            baglanti(bm,r1,r2,r3,r4,
                     r5,r6);
            End;
          3:Begin
            r1:=7;r2:=0;r3:=7;r4:=0;
            r5:=0;r6:=7;menu(i1);
            baglanti(bm,r1,r2,r3,r4,
                     r5,r6);

```

```
                                End;  
                        End;  
                End;  
        6:Begin menu(i1);run;End;  
        7:Begin menu(i1);cikis;end;  
End;  
goto oku;  
son:  
window(1,1,80,25);textbackground(0);textcolor(7);clrscr;  
END.
```



```
{ MENU KONTROLLERININ YAPILDIGI AMPLI.TPU DOSYASI }
```

```
unit ampli;
interface
uses crt;
var
ekran          :array [1..4000] of byte absolute
                $b800:0;
dos             :file;
doso:text;
ch1,ch2:char;
i,i1,i2,j1,j2:byte;

function FileExists(FileName: string): Boolean;
function paralel(p1,p2:real):real;
function yuvarla(sayi:real):real;
procedure imlec;
procedure pencere(sut1,sat1,sut2,sat2:byte);
procedure pencerel(sut1,sat1,sut2,sat2:byte);
procedure menu(i1:byte);
procedure Dosya(dm,r1,r2,r3,r4,r5,r6,r7,r8:byte);
procedure konu(km,r1,r2,r3,r4:byte);
procedure frekans(fm,r1,r2,r3,r4,r5,r6:byte);
procedure eleman(em,r1,r2,r3,r4,r5,r6:byte);
procedure baglanti(bm,r1,r2,r3,r4,r5,r6:byte);
procedure run;
procedure tablo1(vcc,avs,rs,r1,zi,z0,b0min,b0m,bdcmin,bd
                cm,vcb,cmu,wl,wh,rbb,rd,rx,v0,ice0,r0,
                b:real);
procedure tablo2(vcc,r11,r21,rc,re,rs,r1,b0,bdc,c1,c2,c3
                ,ce,cp,cmu,rbb,r0,rd,rx,v0,ice0,b
                :real);
procedure tablo3(vdd,r11,r21,rd,rss,rs,r1,vp,idss,vt,
                kk1,c1,c2,c3,css,cgd,cgs,e,b,rk:real);
procedure tablo4(vdd,avs,zi,z0,wl,wh,cgd,rs,r1,vp,idss,
                vt,kk1,e,b: real);
procedure cikis;
procedure menuk;
procedure yaz(deg:real);
procedure editor(var
                vcc,vdd,r11,r21,rc,rd,r1,re,rss,rs,bdc,
                b0,vp,vt,idss,kk1,c1,
                c2,c3,ce,css,cp,cmu,cgs,cgd,ice0,v0,rx,
                r0,avs,zi,wl,wh,rbb,
                vcb,z0,b0min,b0m,bdcmin,bdcm:real;
                var k,k1:integer; var
                e,b,rk,ins,ilkchar:byte; var
                ch1,ch2:char);
```

```
implementation
```

```
function FileExists(FileName: string): Boolean;
begin
    {$I-}
```

```

Assign(doso, FileName);
Reset(doso);
{$I+}
if IOResult = 0 then begin
  close(doso);fileexists:=true;end
  else fileexists:=false;
end;
function paralel(p1,p2:real):real;
Begin
paralel:=p1*p2/(p1+p2);
end;
function yuvarla(sayi:real):real;
begin
yuvarla:=int(sayi)+round(frac(sayi)*100)/100;
end;
procedure imlec;
begin
window(79,23,79,23);textbackground(0);textcolor(0);
clrscr;
textbackground(7);
end;
procedure yaz(deg:real);
var
s:string[20];
s1:string[20];
k1,k2,k3:integer;
begin
if deg<=9999 then write(deg:4:2) else
begin
str(deg,s);write(copy(s,2,2));
s1:=copy(s,4,1);val(s1,k2,k1);s1:=copy(s,5,1);
val(s1,k3, k1);
if k3>=5 then k2:=k2+1;
write(k2,copy(s,14,4));
end;
end;
procedure pencere(sut1,sat1,sut2,sat2:byte);
begin
textcolor(0);textbackground(7);
window(sut1,sat1,sut2,sat2);clrscr;
window(sut1,sat1,sut2,sat2+1);
write('┌');repeat write('=');until wherex=sut2-sut1+1;
write('┐');gotoxy(1,wherey-1);
repeat gotoxy(1,wherey+1);write('┐');
until wherey=sat2-sat1;gotoxy(1,wherey+1);
write('└');repeat gotoxy(wherex,sat2);write('=');
until wherex=sut2-sut1+1;
gotoxy(sut2-sut1+1,sat2-sat1+1);write('└');
gotoxy(sut2-sut1+1,2);
repeat gotoxy(sut2-sut1+1,wherey);write('└');
until wherey=sat2-sat1+1;
window(sut1+1,sat1+1,sut2-1,sat2-1);
end;
procedure pencere1(sut1,sat1,sut2,sat2:byte);

```

```

begin
window(3,3,77,23);clrscr;
textcolor(0);textbackground(7);
window(sut1-1,sat1,sut2+1,sat2);clrscr;
window(sut1,sat1,sut2,sat2+1);
write('┌');repeat write('-');until wherex=sut2-sut1+1;
write('┐');gotoxy(1,wherex-1);
repeat gotoxy(1,wherex+1);write('|');
until wherex=sat2-sat1;gotoxy(1,wherex+1);
write('└');repeat gotoxy(wherex,sat2-1);write('-');
until wherex=sut2-sut1+1;
gotoxy(sut2-sut1+1,sat2-sat1+1);write('┘');
gotoxy(sut2-sut1+1,2);
repeat gotoxy(sut2-sut1+1,wherex);write('|');
until wherex=sat2-sat1+1;
end;
Procedure menu(il:byte);
begin
assign(dos,'ammono.scr');
reset(dos,4000);
blockread(dos,ekran[1],1);close(dos);
window(1,1,80,24);textbackground(0);textcolor(15);
Case il of
  1:Begin Gotoxy(3,1);Write('D');
textcolor(15);Write('osya ');end;
  2:Begin Gotoxy(10,1);Write('K');
textcolor(15);Write('onu ');end;
  3:Begin Gotoxy(16,1);Write('F');
textcolor(15);Write('rekans seçimi ');end;
  4:Begin Gotoxy(32,1);Write('E');
textcolor(15);Write('leman seçimi ');end;
  5:Begin Gotoxy(47,1);Write('B');
textcolor(15);Write('aglantı şekli ');end;
  6:Begin Gotoxy(63,1);Write('R');
textcolor(15);Write('un ');end;
  7:Begin Gotoxy(68,1);Write('C');
textcolor(15);Write('ikis ');end;
End;
imlec;
End;
Procedure Dosya(dm,r1,r2,r3,r4,r5,r6,r7,r8:byte);
Begin
  if dm=1 then begin
    Window(1,1,80,1);
    textcolor(15);TextBackGround(0);
    Gotoxy(3,1);Write(' D');textcolor(15);Write('osya ');
    pencere1(3,2,11,7);
    textcolor(15);TextBackGround(r1);
    Gotoxy(2,2);Write(' A');TextColor(r2);Write('ç ');
    textcolor(15);TextBackGround(r3);
    Gotoxy(2,3);Write(' S');TextColor(r4);Write('akla ');
    textbackground(r5);textcolor(r6);gotoxy(2,4);
    write(' S');
    textcolor(15);gotoxy(4,4);Write('i');TextColor(r6);

```



```

    Write('1 '); textcolor(15);textbackground(r7);
    gotoxy(2,5);write(' B');textcolor(r8);write('ilgi ');
    imlec;
    end;
End;
Procedure Konu(km,r1,r2,r3,r4:byte);
Begin
    if km=1 then begin
        Window(1,1,80,1);
        textcolor(15);TextBackGround(0);
        Gotoxy(10,1);Write(' K');textcolor(15);Write('onu ');
        pencere1(10,2,20,5);
        textcolor(15);TextBackGround(r1);
        Gotoxy(2,2);
        Write(' A');TextColor(r2);Write('naliz ');
        textcolor(15);TextBackGround(r3);
        Gotoxy(2,3);
        Write(' T');TextColor(r4);Write('asarım ');
        imlec;
        end;
    End;
Procedure Frekans(fm,r1,r2,r3,r4,r5,r6:byte);
Begin
    if fm=1 then begin
        Window(1,1,80,24);textcolor(15);TextBackGround(0);
        Gotoxy(16,1);
        Write(' F');textcolor(15);Write('rekans Seçimi ');
        pencere1(16,2,33,6);
        textcolor(15);TextBackGround(r1);
        Gotoxy(2,2);
        Write(' O');TextColor(r2);Write('rta frekans ');
        textcolor(15);TextBackGround(r3);
        Gotoxy(2,3);
        Write(' A');TextColor(r4);Write('lçak frekans ');
        textcolor(15);TextBackGround(r5);
        Gotoxy(2,4);
        Write(' Y');TextColor(r6);Write('üksek frekans ');
        imlec;
        end;
    End;
Procedure Eleman(em,r1,r2,r3,r4,r5,r6:byte);
Begin
    if em=1 then begin
        Window(1,1,80,24);textcolor(15);TextBackGround(0);
        Gotoxy(32,1);
        Write(' E');textcolor(15);Write('leman seçimi ');
        pencere1(32,2,65,6);
        textcolor(15);TextBackGround(r1);
        Gotoxy(2,2);Write(' B');TextColor(r2);
        Write('ipolar alan etkili transistör ');
        textcolor(15);TextBackGround(r3);
        Gotoxy(2,3);Write(' J');TextColor(r4);
        Write('fet veya aritılmış mosfet ');
        textcolor(15);TextBackGround(r5);
    end;
End;

```

```

Gotoxy(2,4);Write('A');TextColor(r6);
Write('rttirilmis mosfet');
imlec;
end;
End;
Procedure Baglanti(bm,r1,r2,r3,r4,r5,r6:byte);
Begin
  if bm=1 then begin
    Window(1,1,80,24);textcolor(15);TextBackGround(0);
    Gotoxy(47,1);Write(' B');textcolor(15);
    Write('aglantı sekli ');
    pencere1(47,2,67,6);
    TextColor(r2);TextBackGround(r1);
    Gotoxy(2,2);
    Write(' Ortak ');textcolor(15);write('e');
    TextColor(r2);Write('metörlü ');
    TextColor(r4);TextBackGround(r3);
    Gotoxy(2,3);
    Write(' Ortak ');textcolor(15);Write('b');
    TextColor(r4);Write('azlı ');
    TextColor(r6);TextBackGround(r5);
    Gotoxy(2,4);
    Write(' Ortak ');textcolor(15);Write('k');
    TextColor(r6);Write('ollektörlü ');
    imlec;
    end;
  end;
End;
Procedure Run;
Begin
  Window(1,1,80,24);textcolor(15);TextBackGround(0);
  Gotoxy(63,1);textcolor(15);
  Write(' R');textcolor(15);Write('un ');
  imlec;
  end;
Procedure Cikis;
Begin
  Window(1,1,80,24);textcolor(15);TextBackGround(0);
  Gotoxy(68,1);textcolor(15);
  Write(' Ç');textcolor(15);Write('ıkis ');
  imlec;
  end;
procedure tablol(vcc,avs,rs,r1,zi,z0,b0min,b0m,bdcmin,bd
cm,vcb,cmu,wl,wh,rbb,rd,rx,v0,ice0,r0,b
:real);
begin
  window(3,2,77,25);textcolor(15);
  gotoxy(1,1);
  write('_____');
  gotoxy(1,2);
  write(' | VCC(V)      =      | Avs      ≥      | ');
  gotoxy(1,3);
  write('_____');
  gotoxy(1,4);
  write(' | RS(kΩ)      =      | RL(kΩ)      =      | ');

```

```

gotoxy(1,5);
write('_____');
gotoxy(1,6);
write('Zi(kΩ) ≥ Zo(kΩ) ≥ ');
gotoxy(1,7);
write('_____');
gotoxy(1,8);
write('βo(min) = βo(max) = ');
gotoxy(1,9);
write('_____');
gotoxy(1,10);
write('βdc(min) = βdc(max) = ');
gotoxy(1,11);
write('_____');
gotoxy(1,12);
write('Vcb(V) = Cp(pF) = ');
gotoxy(1,13);
write('_____');
gotoxy(1,14);
write('WL(rad/s) ≤ WH(Mrad/s) ≥ ');
gotoxy(1,15);
write('_____');
gotoxy(1,16);
write('RBB(kΩ) = rd(kΩ) = ');
gotoxy(1,17);
write('_____');
gotoxy(1,18);
write('rx(kΩ) = Vo(V) = ');
gotoxy(1,19);
write('_____');
gotoxy(1,20);
write('ICE0(nA) = R0(kΩ) = ');
gotoxy(1,21);
write('_____');
gotoxy(1,22);
write('DEGER GIRISI TAMAM ');
gotoxy(1,23);
write('_____');
if b=2 then begin gotoxy(5,4);write('Ω ');
                  gotoxy(12,6);write('≤');
                  gotoxy(5,6);write('Ω ');
                  end;
if b=3 then begin gotoxy(24,6);write('Ω ');
                  gotoxy(31,6);write('≤');
                  gotoxy(31,2);write('=');
                  end;
gotoxy(2,24);textbackground(7);textcolor(0);
write('< ESC > iPTAL'); textbackground(0);textcolor(7);
gotoxy(13,2);yaz(vcc);gotoxy(32,2);
if b<>3 then yaz(avs);
gotoxy(13,4);if b<>2 then yaz(rs) else yaz(rs*1000);
gotoxy(32,4);yaz(r1);
gotoxy(13,6);if b<>2 then yaz(zi) else yaz(zi*1000);
gotoxy(32,6);if b<>3 then yaz(z0) else yaz(z0*1000);

```

```

gotoxy(13,8);yaz(b0min);gotoxy(32,8);yaz(b0m);
gotoxy(13,10);yaz(bdcmin);gotoxy(32,10);yaz(bdcm);
gotoxy(13,12);yaz(vcb);gotoxy(32,12);yaz(cmu*1E+12);
gotoxy(13,14);if w1<1E+06 then yaz(w1)
                        else yaz(w1/1E+06);
gotoxy(32,14);yaz(wh/1E+06);
gotoxy(13,16);yaz(rbb);gotoxy(32,16);
if rd<>-1 then yaz(rd)
else write('∞');
gotoxy(13,18);yaz(rx*1000);gotoxy(32,18);yaz(v0);
gotoxy(13,20);yaz(ice0*1E+09);gotoxy(32,20);
if r0<>-1 then yaz(r0)
else write('∞'); gotoxy(11,22);textcolor(7);write('D');
gotoxy(13,2);
end;
procedure tablo2(vcc,r11,r21,rc,re,rs,r1,b0,bdc,c1,c2,c3
                ,ce,cp,cmu,rbb,r0,rd,rx,v0,ice0,b
                :real);
begin
window(3,2,77,25);textcolor(15);
gotoxy(1,1);
write('_____');
gotoxy(1,2);
write('VCC(V)      =      | R1(kΩ)      =      |');
gotoxy(1,3);
write('_____');
gotoxy(1,4);
write('R2(kΩ)      =      | RC(kΩ)      =      |');
gotoxy(1,5);
write('_____');
gotoxy(1,6);
write('RE(kΩ)      =      | RS(kΩ)      =      |');
gotoxy(1,7);
write('_____');
gotoxy(1,8);
write('RL(kΩ)      =      | βo      =      |');
gotoxy(1,9);
write('_____');
gotoxy(1,10);
write('βdc      =      | C1(μF)      =      |');
gotoxy(1,11);
write('_____');
gotoxy(1,12);
write('C2(μF)      =      | CE(μF)      =      |');
gotoxy(1,13);
write('_____');
gotoxy(1,14);
write('Cp(pF)      =      | Cμ(pF)      =      |');
gotoxy(1,15);
write('_____');
gotoxy(1,16);
write('RBB(kΩ)      =      | R0(kΩ)      =      |');
gotoxy(1,17);
write('_____');

```

```

gotoxy(1,18);
write('rd(kΩ)      =      |rx(kΩ)      =      |');
gotoxy(1,19);
write('_____|_____');
gotoxy(1,20);
write('Vo(V)      =      |ICE0(nA)      =      |');
gotoxy(1,21);
write('_____|_____');
gotoxy(1,22);
write('DEGER GIRISI TAMAM      |');
gotoxy(1,23);
write('_____|_____');
if b=2 then begin gotoxy(22,12);write('3');
                  gotoxy(24,6);write('Ω');
                  end;
gotoxy(2,24);textbackground(7);textcolor(0);
write('< ESC > iPTAL'); textbackground(0);textcolor(7);
gotoxy(13,2);yaz(vcc);gotoxy(32,2);yaz(r11);
gotoxy(13,4);yaz(r21);gotoxy(32,4);yaz(rc);
gotoxy(13,6);yaz(re);gotoxy(32,6);
if b<>2 then yaz(rs)
      else yaz(rs*1000);
gotoxy(13,8);yaz(r1);gotoxy(32,8);yaz(b0);
gotoxy(13,10);yaz(bdc);gotoxy(32,10);yaz(C1*1E+06);
gotoxy(13,12);yaz(C2*1E+06);gotoxy(32,12);
if b=1 then yaz(c1*1E+06)
      else if b=2 then yaz(c3*1E+06) else yaz(0);
gotoxy(13,14);yaz(cp*1E+12);gotoxy(32,14);
yaz(cmu*1E+12); gotoxy(13,16);yaz(rbb);gotoxy(32,16);
if r0<>-1 then yaz(r0)
      else write('∞');
gotoxy(13,18);if rd<>-1 then yaz(rd) else write('∞');
gotoxy(32,18);yaz(rx*1000);
gotoxy(13,20);yaz(v0);gotoxy(32,20);yaz(ice0*1E+09);
gotoxy(11,22);textcolor(7);write('D');
gotoxy(13,2);
end;
procedure
tablo3(vdd,r11,r21,rd,rss,rs,r1,vp,idss,vt,kk1,c1,c2,c3,
      css,cgd,cgs,e,b,rk:real);
begin
if e=2 then if r11=0 then rk:=1 else rk:=0;
if e=3 then if r21=0 then rk:=1 else rk:=0;
if rd=-1 then rd:=0;
window(3,3,77,25);textcolor(15);
gotoxy(1,1);
write('_____|_____');
gotoxy(1,2);
write('Vdd(V)      =      |R1(kΩ)      =      |');
gotoxy(1,3);
write('_____|_____');
gotoxy(1,4);
write('R2(kΩ) =      |RD(kΩ) =      |');
gotoxy(1,5);

```

```

write(' ');
gotoxy(1,6);
write(' | RSS(k $\Omega$ )      =      | RS(k $\Omega$ )      =      | ');
gotoxy(1,7);
write(' ');
gotoxy(1,8);
write(' | RL(k $\Omega$ )       =      | Vp(V)        =      | ');
gotoxy(1,9);
write(' ');
gotoxy(1,10);
write(' | IDSS(mA)      =      | C1( $\mu$ F)      =      | ');
gotoxy(1,11);
write(' ');
gotoxy(1,12);
write(' | C2( $\mu$ F)        =      | CSS( $\mu$ F)     =      | ');
gotoxy(1,13);
write(' ');
gotoxy(1,14);
write(' | Cgd(pF)       =      | Cgs(pF)     =      | ');
gotoxy(1,15);
write(' ');
gotoxy(1,16);
write(' |          DEGER GIRISI TAMAM          | ');
gotoxy(1,17);
write(' ');
if e=3 then begin gotoxy(22,8);write('T');
                  gotoxy(2,10);write('K(mA/V2)');
                  end;
if b=2 then begin gotoxy(24,6);write('Ω ');
                  gotoxy(22,12);write('3( $\mu$ F) ');
                  end;
gotoxy(2,23);textbackground(7);textcolor(0);
write('< ESC > iPTAL'); textbackground(0);textcolor(7);
gotoxy(13,2);yaz(vdd);gotoxy(32,2);
if (e=2) and (rk=1) then write('∞')
                  else yaz(r11);
gotoxy(13,4);if (e=3) and (rk=1) then write('∞')
                  else yaz(r21);
gotoxy(32,4);if rd<>-1 then yaz(rd) else yaz(0);
gotoxy(13,6);yaz(rss);gotoxy(32,6);
if b<>2 then yaz(rs)
                  else yaz(rs*1000);
gotoxy(13,8);yaz(r1);gotoxy(32,8);
if e=2 then yaz(vp)
                  else yaz(vt);
gotoxy(13,10);if e=2 then yaz(idss)
                  else yaz(kk1);
                  gotoxy(32,10);yaz(C1*1E+06);
gotoxy(13,12);yaz(C2*1E+06);gotoxy(32,12);
if css=-1 then write('∞')
                  else if b<>2 then yaz(css*1E+06)
                  else yaz(c3*1E+06);
gotoxy(13,14);yaz(cgd*1E+12);gotoxy(32,14);
yaz(cgs*1E+12 ); gotoxy(11,16);textcolor(7);write('D');

```

```

gotoxy(13,2);
end;
procedure
tablo4(vdd,avs,zi,z0,wl,wh,cgd,rs,rl,vp,idss,vt,ttl,e,b
      :real);
begin
window(3,3,77,25);textcolor(15);
gotoxy(1,1);
write(' ');
gotoxy(1,2);
write(' | Vdd(V)      =      | Avs      ≥      | ');
gotoxy(1,3);
write(' ');
gotoxy(1,4);
write(' | Zi(kΩ)      ≥      | Zo(kΩ)      ≥      | ');
gotoxy(1,5);
write(' ');
gotoxy(1,6);
write(' | RS(kΩ)      =      | RL(kΩ)      =      | ');
gotoxy(1,7);
write(' ');
gotoxy(1,8);
write(' | Vp(V)      =      | IDSS(mA)  =      | ');
gotoxy(1,9);
write(' ');
gotoxy(1,10);
write(' | WL(rad/s) ≤      | WH(Mrad/s) ≥      | ');
gotoxy(1,11);
write(' ');
gotoxy(1,12);
write(' | Cgd(pF)      =      | DEGER GIRISI TAMAM | ');
gotoxy(1,13);
write(' ');
if e=3 then begin gotoxy(3,8);write('T');
                  gotoxy(21,8);write('K(mA/V²)');
                  end;
if b=2 then begin gotoxy(5,4);write('Ω ');
                  gotoxy(12,4);write('≤');
                  gotoxy(5,6);write('Ω ');
                  end;
if b=3 then begin gotoxy(24,4);write('Ω ');
                  gotoxy(31,4);write('≤');
                  end;
gotoxy(2,23);textbackground(7);textcolor(0);
write('< ESC > iptal'); textbackground(0);textcolor(7);
gotoxy(13,2);yaz(vdd);gotoxy(32,2);
if b<>3 then yaz(avs);
gotoxy(13,4);if b<>2 then yaz(zi)
                else yaz(zi*1000);
                gotoxy(32,4);if b<>3 then yaz(z0)
                                else
yaz(z0*1000);
gotoxy(13,6);if b<>2 then yaz(rs)
                else

```



```

yaz(rs*1000);gotoxy(32,6);yaz(r1);
gotoxy(13,8);if e=2 then yaz(vp)
                        else yaz(vt);
gotoxy(32,8);if e=2 then yaz(idss)
                        else yaz(kk1);
gotoxy(13,10);yaz(w1);gotoxy(32,10);yaz(wh/1E+06);
gotoxy(13,12);yaz(Cgd*1E+12);
gotoxy(21,12);textcolor(7);write('D');
gotoxy(13,2);
end;
procedure editor(var vcc,vdd,r11,r21,rc,rd,r1,re,rss,rs,
                  bdc, b0,vp,vt,idss,kk1,c1,c2,c3,ce,css,
                  cp,cmu,cgs,cgd,ice0,v0,rx,r0,avs,zi,w1,
                  wh,rbb,vcb,z0,b0min,b0m,bdcmin,bdcm
                  :real;
                  var k,k1:integer;
                  var e,b,rk,ins,ilkchar:byte;
                  var ch1,ch2:char);
var
str1:string[40];sol:string[1];
ch,j1,x1,x2:byte;
Begin
if (e=1) and (k=2) then
tablo1(vcc,avs,rs,r1,zi,z0,b0min,b0m,bdcmin,bdcm,vcb,cmu
,w1,wh,rbb,rd,rx,v0,ice0,r0,b);
if (e=1) and (k=1) then
tablo2(vcc,r11,r21,rc,re,rs,r1,b0,bdc,c1,c2,c3,ce,cp,cmu
,rbb,r0,rd,rx,v0,ice0,b);
if (e<>1) and (k=1) then
tablo3(vdd,r11,r21,rd,rss,rs,r1,vp,idss,vt,kk1,c1,c2,c3,
css,cgd,cgs,e,b,rk);
if (e<>1) and (k=2) then
tablo4(vdd,avs,zi,z0,w1,wh,cgd,rs,r1,vp,idss,vt,kk1,e,b)
; j1:=1;str1:='';x1:=1;sol:='0';ch:=0;ilkchar:=0;
repeat
textcolor(0);textbackground(7);
i:=wherex;x2:=wherey;
if e=1 then gotoxy(60,24) else gotoxy(60,23);
if ins=0 then write('INS') else write(' ');
textcolor(7);textbackground(0);gotoxy(i,x2);
ch1:=#0;ch2:=#0;
ch1:=readkey;if keypressed Then ch2:=readkey;
if (ch1=#0) then if ((wherey<=22) or (e<>1)) and
((wherey<=16) or (k=2) or (e=1)) and
((wherey<=12) or (k=1) or (e=1)) then
case ch2 of
#59:begin
ch1:=char(236);ch2:=char(0);end;
#72:if (wherey<>2) and
(not(wherey in[6,14]) or
(wherey<20) or (b<>3))
then
begin
ilkchar:=0;ch:=0;x1:=1;j

```



```

        1:=1;str1:='';
        if wherex<19 then
            gotoxy(13,wherey-2)
        else
            gotoxy(32,wherey-2);
        end
    else write(#7);
        #75:if (j1>1) and (wherex in
            [13..19,32..38]) then
            begin
                j1:=j1-1;
                if wherex in [13,32] then
                    begin
                        sol:='0';
                        if x1>1 then x1:=x1-1;
                        end
                    else
                        sol:='1';
                    end;
            #77:if (j1<=length(str1)) and
                (wherex in [13..19,32..38])
                then
                begin
                    j1:=j1+1;
                    if (wherex in [19,38]) and
                        (x1<=length(str1)-7) then
                        x1:=x1+1;
                    if not (wherex in [19,38])
                    then
                        gotoxy(wherex+1,wherey);
                    end;
            #80:if ((not(wherey in [2,12]))
                or (wherex<20) or (b<>3))
                and (((wherey<>22) and
                    (e=1)) or
                    ((wherey<>16) and (e<>1)
                    and (k=1)) or
                    ((wherey<>12) and (e<>1)
                    and (k<>1))) then
                begin
                    ilkchar:=0;ch:=0;x1:=1;j1
                    :=1;str1:='';
                    if wherex<19 then
                        gotoxy(13,wherey+2)
                    else
                        gotoxy(32,wherey+2);
                    end
                    else write(#7);
            #83:delete(str1,j1,1);
            #71:begin
                j1:=1;x1:=1;
                if wherex<20 then
                    gotoxy(13,wherey)

```

```

else
    gotoxy(32,wherey);
    end;
#79:begin
    j1:=length(str1)+1;
    if j1>7 then
    begin
        j1:=j1-
        1;x1:=j1-7;end else x1:=1;
        if wherex in [13..20] then
        if length(str1)<8 then
gotoxy(14+length(str1)-x1,wherey)

        else begin

            gotoxy(19,wherey);

            end;
            if wherex in [32..39] then
            if length(str1)<8 then
gotoxy(33+length(str1)-x1,wherey)

            else begin

                gotoxy(38,wherey);

                ch:=1;
                end;
                end;
                #82:if ins=0 then ins:=1 else
                ins:=0;
                end else write(#7);
                if (ch1=#8) and (ch2=#0) then
if ((wherex in [14..19]) or (wherex in [33..38])) and
((wherey<>22) or (e<>1)) and ((wherey<>16) or
(k=2) or (e=1)) and
((wherey<>12) or (k=1) or (e=1)) then
begin
    if j1>1 then
    j1:=j1-1;gotoxy(wherex-1,wherey);
    delete(str1,j1,1);
    end;
if (ch1 in [#9,#13,#68,#100]) and (ch2=#0) then
if (wherex<20) and ((wherey<>22) or (e<>1)) and
((wherey<>16) or (k=2) or (e=1)) then begin
    if str1<>' ' then
    if e=1 then case k of
        1:case wherey of
2:val(str1,vcc,k1);
4:val(str1,r21,k1);
6:val(str1,re,k1);
8:val(str1,rl,k1);
10:val(str1,bdc,k1);

```

```

12:begin val(str1,c2,k1);c2:=c2*1E-06; end;
14:begin val(str1,cp,k1);cp:=cp*1E-12; End;
16:val(str1,rbb,k1);
18:if str1<>'∞' then val(str1,rd,k1)
else rd:=-1;
20:val(str1,v0,k1);

                                end;
                                2:case wherey of
2:val(str1,vcc,k1);
4:begin val(str1,rs,k1); if b=2 then rs:=rs/1000; end;
6:begin val(str1,zi,k1); If b=2 then zi:=zi/1000; end;
8:val(str1,b0min,k1);
10:val(str1,bdcmin,k1);
12:val(str1,vcb,k1);
14:val(str1,w1,k1);
16:val(str1,rbb,k1);
18:begin val(str1,rx,k1);rx:=rx/1000; end;
20:val(str1,ice0,k1);

                                end;
                                end;
                                if str1<>' ' then
                                if e<>1 then case k of
                                1:case wherey of
2:val(str1,vdd,k1);
4:if str1<>'∞' then begin val(str1,r21,k1);
                        if e=3 then rk:=0;
                        end
                        else rk:=1;
6:val(str1,rss,k1);
8:val(str1,r1,k1);
10:if e=2 then val(str1,idss,k1)
    else val(str1,kk1,k1);
12:begin val(str1,c2,k1);c2:=c2*1E-06;end;
14:begin val(str1,cgd,k1);cgd:=cgd*1E-12;end;
                                end;
                                2:case wherey of
2:val(str1,vdd,k1);
4:begin val(str1,zi,k1);if b=2 then zi:=zi/1000; end;
                                6:begin
val(str1,rs,k1);if b=2 then rs:=rs/1000;end;
8:if e=2 then val(str1,vp,k1)
    else val(str1,vt,k1);
10:val(str1,w1,k1);
12:begin
val(str1,cgd,k1);cgd:=cgd*1E-12;

                                end;
                                end;
                                end;
                                str1:=' ';j1:=1;x1:=1;ch:=0;
                                if ((wherey=12) and (e<>1) and (k=2))
                                then begin

```

```

chl:=chr(255);

gotoxy(21,12);
end
else begin
    if (wherey in [4,12]) and (k=1) and (b=3) then
        gotoxy(13,wherey+2)
    else
        if (wherey=2) and (k=2) and (b=3) then
            gotoxy(13,wherey+2)
        else
            gotoxy(32,wherey);
        end;
    end else
    if (wherex<=39) and ((wherey<>22) or (e<>1)) and
        ((wherey<>16) or (k=2) or (e=1)) or (ord(chl)=9) then
    begin
        if str1<>' ' then
            if e=1 then case k of
                1:case wherey of
                    2:val(str1,r11,k1);
                    4:val(str1,rc,k1);
                    6:begin val(str1,rs,k1);if b=2 then rs:=rs/1000;end;
                    8:val(str1,b0,k1);
                    10:begin val(str1,c1,k1);c1:=c1*1E-06;end;
                    12:begin val(str1,ce,k1);ce:=ce*1E-06;
                        if b=2 then c3:=ce;
                        end;
                    14:begin val(str1,cmu,k1);cmu:=cmu*1E-12;end;
                    16:if str1<>'∞' then val(str1,r0,k1) else r0:=-1;
                    18:begin val(str1,rx,k1);rx:=rx/1000;end;
                    20:val(str1,ice0,k1);end;
                2:case wherey of
                    2:val(str1,avs,k1);
                    4:val(str1,r1,k1);
                    6:begin val(str1,z0,k1);if b=3 then z0:=z0/1000;end;
                    8:val(str1,b0m,k1);
                    10:val(str1,bdcm,k1);
                    12:begin val(str1,cmu,k1);cmu:=cmu*1E-12;end;
                    14:begin val(str1,wh,k1);wh:=wh*1E+06;end;
                    16:val(str1,rd,k1);
                    18:val(str1,v0,k1);
                    20:val(str1,r0,k1);
                end;
            end;
            if str1<>' ' then
                if e<>1 then case k of
                    1:case wherey of
                        2:if

```

str1<>'∞' then begin
val(str1,r11,k1);
if e=2 then rk:=0;
end

```

else rk:=1;

4:val(str1,rd,k1);
6:begin val(str1,rs,k1);if b=2 then rs:=rs/1000;end;
8:if e=2 then val(str1,vp,k1) else val(str1,vt,k1);
10:begin val(str1,cl,k1);cl:=cl*1E-06;end;
12:if str1='∞' then css:=-1 else
      begin
        val(str1,c ss,k1);
        css :=css*1E-06;
        if b=2 then c3:=css;
      end;
14:begin val(str1,cgs,k1);cgs:=cgs*1E-12;end;
      end;
      2:case wherey of

2:val(str1,avs,k1);
4:begin val(str1,z0,k1);if b=3 then z0:=z0/1000;end;
6:val(str1,rl,k1);
8:if e=2 then val(str1,idss,k1)
      else val(str1,kkl,k1);
10:begin val(str1,wh,k1);wh:=wh*1E+06;end;
      end;
      str1:='';j1:=1;x1:=1;ch:=0;
      if (wherey=20) and (e=1) then begin

ch1:=chr(255);gotoxy(11,22);
      end
      else
      if ((wherey=22) and (e=1)) or
        ((wherey=16) and (e<>1) and
        (k=1)) or ((wherey=12) and (e<>1)
        and (k=2)) then begin
        if (ord(ch1)<>13) then gotoxy(13,2);
        end
      else
        if (wherey=14) and (e<>1) and (k=1)
then begin ch1:=chr(255);gotoxy(11,16);end
else gotoxy(13,wherey+2);
      end;
      if (ord(ch1) in [48..57,43,45,46,69,236]) and (ch2=#0)
and ((wherey<>22) or (e<>1)) and ((wherey<>16) or (e=1)
or (k=2)) and (not((wherey=12) and (wherex in [21..38])))
or (e=1) or (k=1)) then
begin
if wherex<13 then gotoxy(13,wherey) else if wherex in
[21..31] then gotoxy(32,wherey);
  if ins=1 then delete(str1,j1,1);
  insert(ch1,str1,j1);ilkchar:=1;
  if (j1>=length(str1)) then j1:=j1+1;
  if (wherex in [19,38]) and (ins=0) and
  (length(str1)>7) then x1:=x1+1;

```

```

        if wherex in [19,38] then ch:=1 else ch:=0;
end

else
if not ((ord(ch1) in [0,8,9,13,27,68,100,236,255]) and
        (ord(ch2) in [0,59,71,72,75,77,79,80,82,83]))
then write(#7);
if ((wherey=22) and (e=1)) or ((wherey=16) and (e<>1)
and (k=1)) or ((wherey=12) and (wherex in [21..38]) and
(e=2) and (k=2)) then
if e=1 then gotoxy(11,22)
        else if k=1 then gotoxy(11,16)
                else gotoxy(21,12);
if ((wherey<>22) or (e<>1)) and ((wherey<>16) or (e=1)
or (k=2)) and
        (not((wherey=12) and (wherex in [21..38])) or (e=1)
or (k=1)) then
if wherex<13 then gotoxy(13,wherey) else if wherex in
[21..31] then gotoxy(32,wherey);
if (ord(ch1) in [0,8,48..57,43,45,46,69,236]) and
        (ord(ch2) in [0,59,71,72,75,77,79,80,82,83]) and
        (ilkchar=1) then
begin
i:=wherex;
if wherex in [13..19] then begin write(' ':20-
wherex);gotoxy(i,wherey);end;
if wherex in [32..38] then begin write(' ':39-
wherex);gotoxy(i,wherey);end;
if wherex<20 then gotoxy(13,wherey) else
gotoxy(32,wherey);
write(copy(str1,x1,7));
end;
if (j1<=length(str1)) or (length(str1)>7) then
gotoxy(i,wherey);
if (wherex in [20,39]) or ((wherex in [19,38]) and
(length(str1)>7)) then ch:=1;
if (wherex in [20,39]) or (sol='1') then begin
                                gotoxy(wherex-
1,wherey);
                                end;
sol:='0';
if (ch1 in ['D','d']) and (ord(ch2)=0) then
if e=1 then gotoxy(11,22)
        else if k=1 then gotoxy(11,16)
                else gotoxy(21,12);
until (((wherey=22) and (e=1)) or ((wherey=16) and
(e<>1) and (k=1)) or ((wherey=12) and (e<>1) and (k=2)
and (wherex>20))) and (ord(ch1)=13) and (ord(ch2)=0) or
((ord(ch1)=27) and (ord(ch2)=0));
End;
Procedure menuk;
Label Basla;
Begin
basla:

```

```
ch1:=readkey;
j1:=Ord(ch1);i2:=1;
If Keypressed Then Begin
    ch2:=readkey;
    j2:=Ord(ch2);
    End;
Case j1 Of
    27:Case j2 Of
        37:i1:=2;
        33:i1:=3;
        18:i1:=4;
        48:i1:=5;
        19:i1:=6;
        46:i1:=7;
    End
    ;
    Else Goto Basla;
End;
End;
begin
end.
```

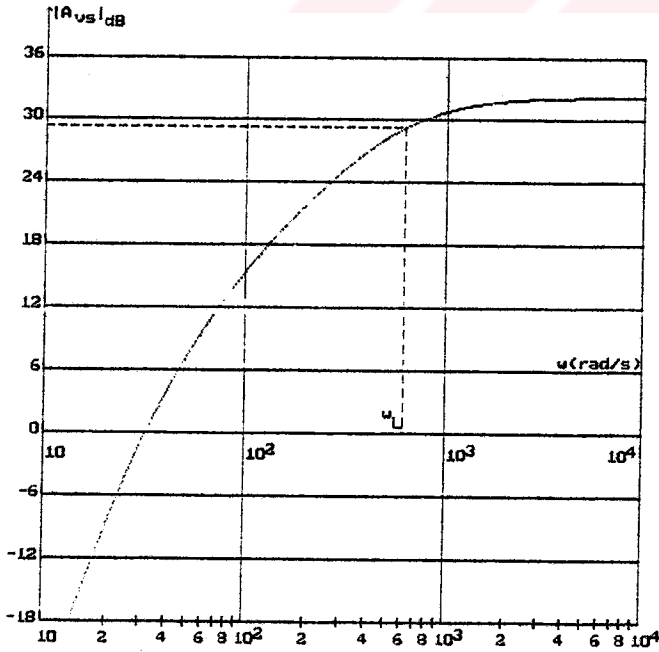
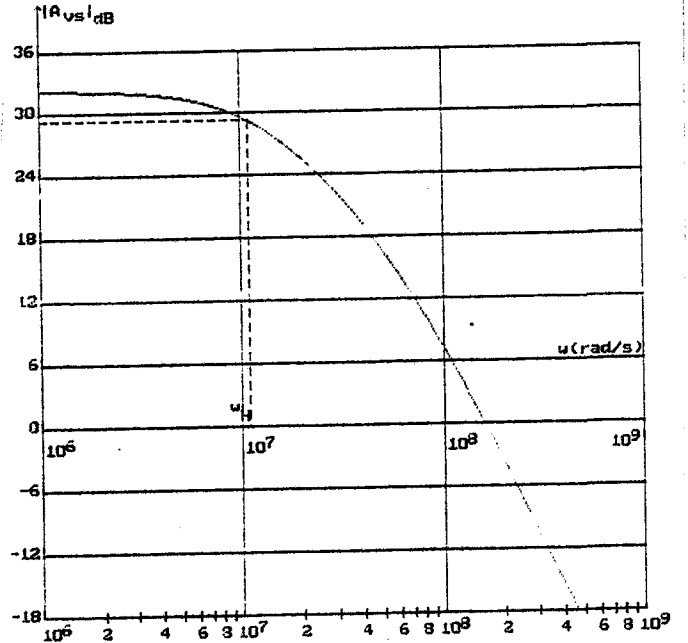
7. BİLGİSAYARDAN ALINAN SONUÇLAR

Bipolar transistörlü ortak emetör bağlantılı amplifikatörün analiziGiriş değerleri

VCC(V)	=15.00	R1(k Ω)	=50.00
R2(k Ω)	=25.00	RC(k Ω)	=5.00
RE(k Ω)	=5.00	RS(k Ω)	=1.00
RL(k Ω)	=2.00	β_o	=350.00
β_{dc}	=350.00	C1(μ F)	=3.00
C2(μ F)	=8.00	CE(μ F)	=50.00
Cp(pF)	=20.00	C μ (pF)	=2.00
RBB(k Ω)	=0.00	R0(k Ω)	= ∞
rd(k Ω)	= ∞	rx(k Ω)	=0.00
Vo(V)	=0.65	ICE0(nA)	=0.00

Orta frekans analizi

$r_{th}=16.67 \text{ k}\Omega$
 $v_{th}=5.00 \text{ V}$
 $V_{ce}=6.39 \text{ V}$
 $I_c=0.86 \text{ mA}$
 $g_m=33.45 \text{ mmho}$
 $r_{\pi}=10.46 \text{ k}\Omega$
 $A_{vi}=-47.78$
 $A_{vs}=-41.35$
 $Z_i=6.43 \text{ k}\Omega$
 $A_i=-153.57$
 $A_p=7337.87$
 $Z_O=5.00 \text{ k}\Omega$

Alçak frekans analiziYüksek frekans analizi

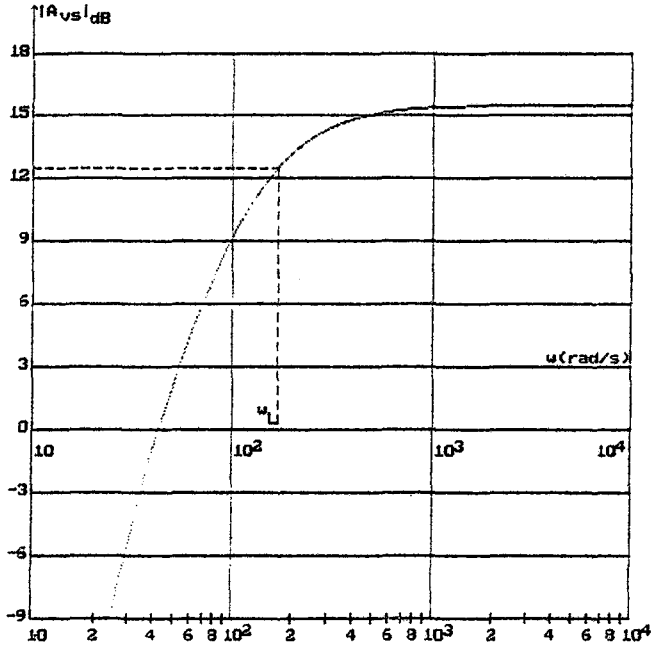
Fetli ortak baz bağlantılı amplifikatörün analizi

Giriş değerleri

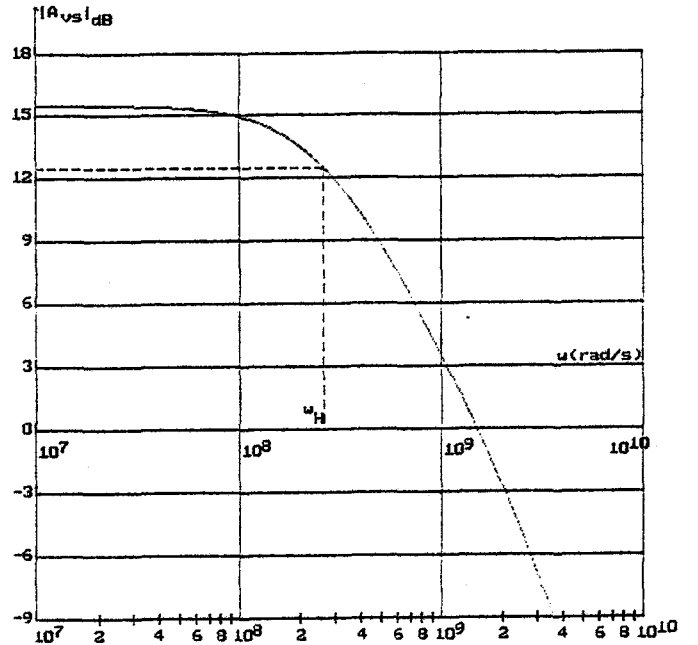
Vdd(V)	=20.00	R1(k Ω)	=1700.00
R2(k Ω)	=300.00	RD(k Ω)	=3.00
RSS(k Ω)	=1.00	RS(Ω)	=50.00
RL(k Ω)	=5.00	Vp(V)	=-3.00
IDSS(mA)	=9.00	C1(μ F)	=25.00
C2(μ F)	=5.00	C3(μ F)	=0.10
Cgd(pF)	=2.00	Cgs(pF)	=5.00

ID =4.00 mA
 VDS=4.00 V
 VGS=-1.00 V
 gm =4.00 mmho
 Avi=7.50
 Avs=6.00
 Zi =200.00 Ω
 Ai =0.30
 Ap =2.25
 ZO =3.00 k Ω

Alçak frekans analizi



Yüksek frekans analizi



Artırılmış mosftli ortak kollektör bağlantılı amplifikatörün analizi

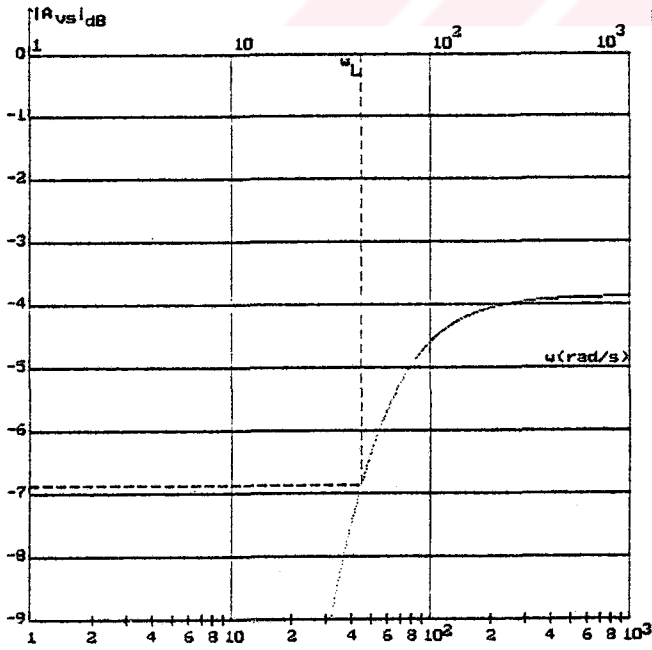
Giriş değerleri

Vdd(V)	=20.00	R1(k Ω)	=1700.00
R2(k Ω)	=300.00	RD(k Ω)	=0.00
RSS(k Ω)	=1.00	RS(k Ω)	=1.00
RL(k Ω)	=5.00	VT(V)	=2.00
K(mA/V ²)	=4.50	C1(μ F)	=0.02
C2(μ F)	=5.00	CSS(μ F)	=0.00
Cgd(pF)	=2.00	Cgs(pF)	=5.00

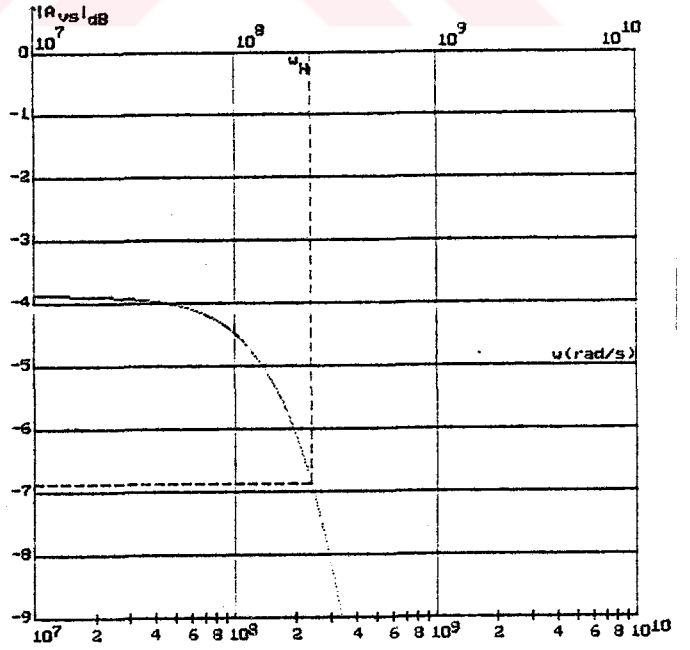
Orta frekans analizi

$I_D=0.52$ mA
 $V_{DS}=19.48$ V
 $V_{GS}=2.48$ V
 $g_m=2.16$ mmho
 $A_{vi}=0.64$
 $A_{vs}=0.64$
 $Z_i=255.00$ k Ω
 $A_i=32.80$
 $A_p=21.09$
 $Z_O=316.23$ Ω

Alçak frekans analizi



Yüksek frekans analizi



Bipolar transistörlü ortak kollektör bağlantılı amplifikatörün tasarımı

Giriş değerleri

Bilgisayar sonuçları

VCC(V)	=15.00	A _{vs}	=
RS(kΩ)	=1.00	RL(kΩ)	=2.00
Z _i (kΩ)	≥15.00	Z _o (Ω)	≤30.00
β _o (min)	=350.00	β _o (max)	=1400.00
β _{dc} (min)	=350.00	β _{dc} (max)	=1080.00
V _{cb} (V)	=4.35	C _μ (pF)	=2.00
WL(rad/s)	≤100.00	WH(Mrad/s)	≥400.00
R _{BB} (kΩ)	=0.00	r _d (kΩ)	=∞
r _x (kΩ)	=0.00	V _o (V)	=0.65
ICE0(nA)	=0.00	R _O (kΩ)	=∞

V_{cc}=15 V
 R₁ =65.45 kΩ
 R₂ =40.36 kΩ
 R_E =5.00 kΩ
 C₁ =7.794 μF
 C₂ =4.936 μF
 C_μ =2.00 pF
 C_p =30.35 pF
 (f_T = 193.11 MHz)

Jfetli ortak emetör bağlantılı amplifikatörün tasarımı

Giriş değerleri

Bilgisayar sonuçları

Vdd(V)	=20.00	Avs	≥7.00
Zi(kΩ)	≥255.00	Zo(kΩ)	≥3.00
RS(kΩ)	=1.00	RL(kΩ)	=5.00
Vp(V)	=-3.00	IDSS(mA)	=9.00
WL(rad/s)	≤300.00	WH(Mrad/s)	≥40.00
Cgd(pF)	=2.00		

R1 = ∞
 R2 =255.00 kΩ
 RD ≥ 3.00 kΩ
 Rss=320.6 Ω
 C1 =0.260 μF
 C2 =8.333 μF
 C_{ss}=22.89 μF
 Cgd =2.00 pF
 Cgs =12.11 pF

Artırılmış mosfetli ortak baz bağlantılı amplifikatörün tasarımı

Giriş değerleri

Bilgisayar sonuçları

Vdd(V)	=20.00	Avs	≥4.04
Zi(Ω)	≤300.00	Zo(kΩ)	≥3.00
RS(Ω)	=50.00	RL(kΩ)	=5.00
VT(V)	=2.00	K(mA/V ²)	=4.50
WL(rad/s)	≤1000.00	WH(Mrad/s)	≥250.00
Cgd(pF)	=2.00		

R1 =1000.00 kΩ
 R2 =206.06 kΩ
 RD ≥ 3.00 kΩ
 Rss=1.222 kΩ
 C1 =2.857 μF
 C2 =2.500 μF
 C3=66.67 μF
 Cgd =2.00 pF
 Cgs =23.87 pF

KAYNAKLAR

- 1- BAYBURAN, Bahattin, 1988. Turbo Pascal. Bilgehan Basımevi.
- 2- LEBLEBİCİ, D., 1970. Elektronik Devreleri ve Sentezi. İ.T.Ü. İnşaat Fakültesi Matbaası.
- 3- LEBLEBİCİ, D., 1980. Elektronik Devreleri 1. İ.T.Ü. Elektrik Fak. Baskı Atölyesi.
- 4- LEBLEBİCİ, D., 1984. Elektronik Devreler ve Düzenleri. İ.T.Ü. Elektrik.-Elektronik Fak. Baskı Atölyesi.
- 5- PASTACI, H., 1985. Elektronik. Yıldız Üniversitesi Matbaası.



ÖZGEÇMİŞ

10.02.1967 yılında Kilis 'te doğdu. İlköğrenimini İskenderun Barbaros İlkoku-
lu 'nda ortaöğrenimini ise İskenderun Barbaros Lisesi 'nde tamamladı. 1985 yılında girdiği
Yıldız Üniversitesi Mühendislik Fakültesi Elektrik Mühendisliği Bölümünden 1990
yılında mezun oldu. Aynı yıl Yıldız Üniversitesi Fen Bilimleri Enstitüsü Elektrik
Mühendisliği Anabilimdalı 'nda yüksek lisans öğrenimine başladı.

T.C. YÜREKLİ ÖRETEL KURULU
DOĞUM MANTASTON MERKEZİ