

**YILDIZ TEKNİK ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ**

168460

# İNCE CİDARLI MÜTEMADİ KİRİŞLERİN ÇARPILMALI BURULMAYA GÖRE HESABI

İnş. Müh. Tuna ÇİÇEK

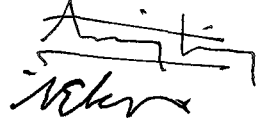
F.B.E İnşaat Mühendisliği Anabilim Dalı Yapı Programında  
Hazırlanan

**YÜKSEK LİSANS TEZİ**

**Tez Danışmanı** : Prof. A Zafer ÖZTÜRK

**Üye** : Prof. İbrahim EKİZ

**Üye** : Prof. Dr. Tuncer ÇELİK



**İSTANBUL, 2005**

## İÇİNDEKİLER

Sayfa

SİMGE LİSTESİ .....	iv
ŞEKİL LİSTESİ .....	v
ÇİZELGE LİSTESİ .....	vi
ÖNSÖZ.....	vii
ÖZET .....	viii
ABSTRACT .....	ix
1. GİRİŞ.....	1
2. TANIMLAR .....	3
2.1 Koordinat Sistemi Ve Dış Tesirler .....	3
3. İNCE CİDARLI AÇIK KESİTLER .....	5
3.1 Kabuller .....	5
3.2 Kesit Değerlerinin Hesabı .....	5
3.2.1 İnce Cidarlı Açık Kesitlerde $\omega$ Birim Çarpılma .....	5
3.2.2 $J_0$ Burulma Mukavemeti .....	6
3.2.3 Başlangıç Koordinatlarının Standartlaştırılması.....	6
3.3 Kesit Tesirleri .....	9
3.4 Çarpılmalı Burulma Diferansiyel Denkleminin Çözümü .....	10
3.5 Sınır ve Geçiş Şartları.....	11
3.6 Çubuk Hesabı .....	12
4. İNCE CİDARLI KAPALI KESİTLER .....	14
4.1 Tek Gözlü Kapalı Kesitlerde Bredt Burulması.....	14
4.2 Çok Gözlü Kapalı Kesitlerde Bredt Burulması .....	16
5. İNCE CİDARLI MÜTEMADİ ÇUBUKLARIN HESABINDA GEÇİŞ MATRİSLERİ METODUNUN KULLANILMASI .....	18
5.1 Geçiş Matrisinin Elde Edilmesi .....	19
5.2 Yük Vektörünün Elde Edilmesi.....	22
5.2.1 Münferit Burulma Momenti .....	22
5.2.2 Münferit Bimoment .....	23
5.2.3 Yayılı Burulma Momenti.....	24
5.3 Ara Mesnetler .....	24
6. İNCE CİDARLI MÜTEMADİ ÇUBUKLARIN HESABININ BİLGİSAYAR PROGRAMI İLE YAPILMASI .....	26

6.1	TWS Programının Tanıtımı .....	26
6.2	TWS Programıyla Sistem Tanımı ve Analizi .....	26
6.3	TWS Programının Çözüm Yöntemi .....	32
6.3.1	TWS Programı Enkesit Tipleri .....	32
6.3.2	TWS Programı Analiz Akış Diyagramları .....	37
KAYNAKLAR.....		50
EKLER .....		51
Ek 1 Sayısal Örnek – TWS Programı Analiz Sonuçları.....		52
Ek 2 TWS Programı Analiz Kodları .....		76
ÖZGEÇMİŞ.....		122



## SİMGE LİSTESİ

A	Kesit Alanı
E	Elastisite modülü
G	Kayma modülü
$x, y, z$	Dik koordinatlar
$s, n$	Kesit yüzeyinin koordinatları
$\omega(s)$	Burulmada çarpılma koordinatları
$r_t(s)$	Kesit orta yüzeyine çizilen teğetin dönme merkezinden mesafesi
L	Çubuk uzunluğu
$t(s)$	Et kalınlığı
$F_{xx}$	x eksenine göre kesit atalet momenti
$F_{yy}$	y eksenine göre kesit atalet momenti
$F_{xy}$	kesit çarpım atalet momenti
$F_{\omega x}$	Sektörel çarpım atalet momenti
$F_{\omega y}$	Sektörel çarpım atalet momenti
$F_{\omega\omega}$	Çarpılma mukavemeti
$F_x$	Statik moment
$F_y$	Statik moment
$F_\omega$	Statik moment
$J_o, J_p$	St. Vénant burulma sabiti
$J_t$	Bredt burulma sabiti
$J_b = J_p + J_t$	Toplam burulma sabiti
B	Bimoment
$M_{BP}$	St. Vénant (Bredt) burulma momenti (primer)
$M_{BS}$	Çarpılmadan doğan ilave burulma momenti (sekonder)
$M_B$	Toplam burulma momenti
$M_t$	Tekil dış burulma momenti
$m_t$	düzgün yayılı dış burulma momenti
T	Kayma akımı
$\sigma_z$	Normal gerilme
$\tau_p$	Primer kayma gerilmesi
$\tau_{sk}$	Sekonder kayma gerilmeleri
$\Psi$	Kayma gerilmesi dağılım (burulma) fonksiyonu
$\vartheta$	Çarpılma ölçüsü
$\varphi$	Çubuk dönmesi
$\varphi'$	izafi çubuk dönmesi
$\phi$	Gerilme fonksiyonu
S	Ağırlık merkezi
D	Kayma merkezi
$A_g$	Kapalı kesitlerde gözlerin alanı
$x_d, y_d$	Kayma merkezinin koordinatları
$d_x, d_y$	Kayma merkezinin seçilen ilk eksen takımına nazaran koordinatları
$x_b, y_b$	Burulma merkezinin koordinatları
k	Kısaltma katsayısı
$A_d$	Diferansiyel geçiş matrisi
$V(z)$	Durum vektörü
$V(0) = V_X$	Başlangıç değerleri vektörü
$VB(z)$	Geçiş Matrisi
$K(z)$	Süreksizlik vektörü
$VP(z)$	Yük vektörü

## ŞEKİL LİSTESİ

Şekil 2.1 – Koordinatlar sistemi (Öztürk, 1979) .....	3
Şekil 2.2 – Kesit tesirleri (Öztürk, 1979) .....	3
Şekil 2.3 – Dış tesirler (Öztürk, 1979) .....	4
Şekil 2.4 – Lokal koordinatlar ve yer değiştirmeler (Öztürk, 1979) .....	4
Şekil 2.5 – Burulma momenti ve bimoment pozitif notasyonu (Öztürk, 1979) .....	4
Şekil 3.1 - $\omega$ Birim çarpılma (Gülek, 1985) .....	5
Şekil 3.2 – x,y asal eksenleri (Öztürk, 1979).....	7
Şekil 4.1 – Kapalı kesitin kesilerek açık kesit haline getirilmesi (Öztürk, 1979) .....	15
Şekil 5.1 – Münferit burulma momenti yüklemesi (Öztürk, 1979).....	22
Şekil 5.2 – Münferit bimoment yüklemesi (Öztürk, 1979) .....	23
Şekil 5.3 – Yayılı burulma momenti yüklemesi (Öztürk, 1979) .....	24
Şekil 6.1 – TWS Programı Ana Penceresi.....	27
Şekil 6.2 – Malzeme Bilgileri Tanımlama Formu .....	27
Şekil 6.3 – Sistem Bilgileri Tanımlama Formu .....	28
Şekil 6.4 – Mesnet Şartları Tanımlama Formu.....	28
Şekil 6.5 – Kesit Bilgi Giriş Formu .....	29
Şekil 6.7 – Yükleme Bilgileri Tanımlama Formu .....	29
Şekil 6.8 – Moment Diyagramları Görüntüleme Formu .....	30
Şekil 6.9 – Analiz Sonuçları Seçeneleri Formu.....	30
Şekil 6.10 – Analiz Sonucu Raporu .....	31
Şekil 6.11 – Enkesit gerilme diyagramları seçenek formu.....	31
Şekil 6.12 – Enkesit gerilme diyagramı formu.....	32
Şekil 6.13 – I Kesit bilgileri .....	33
Şekil 6.14 – C Kesit bilgileri .....	33
Şekil 6.15 – C Kesit bilgileri .....	34
Şekil 6.16 – Tek gözlü kapalı kesit bilgileri.....	35
Şekil 6.17 – Çift gözlü kapalı kesit bilgileri.....	36
Şekil 4.10 – “Main” Alt Programı Akış Şeması.....	37
Şekil 4.11 – “Kesit Hesapla” Alt Programı Akış Diyagramı .....	38
Şekil 4.12 – “kapalı_kesit_on_hesap” alt programı akış diyagramı.....	39
Şekil 4.13 – “w_ilk_kapalı” alt programı akış diyagramı .....	40
Şekil 4.14 – “asal_ksen” Alt Programı Akış Diyagramı .....	42
Şekil 4.15 – “VX_MATRIS” alt programı akış diyagramı .....	43
Şekil 4.16 – “SISTEM” alt programı akış diyagramı .....	43
Şekil 4.17 – “VP_yuk_matrisi” alt programının akış diyagramı.....	44
Şekil 4.18 – “K_sureksizlik_matrisi” alt programının akış diyagramı.....	44
Şekil 4.19 – “DENKLEMLER” alt programının akış diyagramı .....	45
Şekil 4.20 – “sistem_cozum” aslt programının akış diyagramı.....	45
Şekil 4.21 – “bilinmeyen_bul” alt programının akış diyagramı.....	46
Şekil 4.22 – “Kesit_Tesirleri” alt programının akış diyagramı .....	47
Şekil 4.23 – “carpilma” alt programının akış diyagramı .....	48
Şekil 4.24 – “kesit_gerilme” alt programının akış diyagramı .....	49
Şekil Ek 1.1 – Birinci örnek sistemi .....	52
Şekil Ek 1.2 – Enkesit bilgileri .....	52
Şekil Ek 1.3 – İkinci örnek sistemi.....	60
Şekil Ek 1.4 – Enkesit Bilgileri .....	60
Şekil Ek 1.5 – Üçüncü örnek sistemi.....	68
Şekil Ek 1.6 – Kesit bilgileri .....	68

## ÇİZELGE LİSTESİ

Çizelge 3.1 – Sınır Şartları (Öztürk, 1979).....	11
Çizelge 3.2 – Geçiş Şartları (Öztürk, 1979) .....	11
Çizelge 3.3 – Başlangıç parametreleri matrisi, $z=0$ için (Öztürk, 1979).....	13
Çizelge 3.4 – Başlangıç parametreleri matrisi, $z=\zeta$ ; $0 < \zeta < z$ için (Öztürk, 1979) .....	13



## ÖNSÖZ

Kesitin, burulma etkisi ve bununla beraber meydana gelen çarpılmasından dolayı oluşan gerilmeler özellikle ince cidarlı kesitler üzerinde ihmal edilemeyecek seviyede önemli etkiler oluşturmaktadır.

Bu çalışmada ince cidarlı mütemadi çubukların çarpılmalı burulmaya göre analizini, geçiş matrisi metodunu kullanarak yapan TWS isimli bir bilgisayar yazılımı geliştirilmiştir.

Çalışmalarım sırasında desteğini esirgemeyen sayın hocam Prof. A. Zafer ÖZTÜRK'e, mühendislik birikimi ve vizyonuyla bana yol gösteren İnş. Müh. İlker DURMUŞOĞLU'na ve hep bir dayanak noktası olarak hissettiğim aileme sonsuz teşekkürlerimi sunarım.



## ÖZET

Özellikle çelik yapılarda kullanılan elemanlarda, cidar kalınlıkları diğler boyutlarına göre çok küçük olan narin kesitlere rastlanmaktadır. Bu tip elemaların analizinde, gerilme hesabı klasik mukavemetteki bağıntılarla yapıldığı takdirde sonuçlar gerçeği yansıtmamakta, normal kuvvet ve bileşik eğilme ifadeleri olayın tamamını kapsamamaktadır.

Yüklerin statik sisteme eksantrik etkimesi durumu burulma etkisini beraberinde getirmektedir. Asıl fark burulma etkisinin gözetilmesiyle kendini göstermektedir. Burulma etkisi altında oluşan deformasyon sonucu, klasik mukavetten farklı olarak kesitler düzlemsellikten çıkmakta, enkesit çarpılması meydana gelmektedir.

Burulma etkisi ve enkesit çarpılması özellikle ince cidarlı elemanlarda ihmal edilemeyecek düzeyde normal gerilme ve kayma gerilmeleri meydana getirmektedir.

Bu çalışma, Prof. A. Zafer ÖZTÜRK'ün doçentlik tezi temel alınarak geliştirilmiş ince cidarlı mütemadi çubukların çarpılmalı burulmaya göre analizini gerçekleştiren bir bilgisayar yazılımını içermektedir. Çalışmanın ilk kısımlarında çarpılmalı burulmanın teorik alt yapısı için gerekli olan bilgiler sözü edilen tezden alınarak aktarılmıştır.

**Anahtar kelimeler:** İnce cidarlı çelik, çarpılmalı burulma, geçiş matrisi metodu.

## **ABSTRACT**

Especially in the items used in steel constructions, cross sections which are not as thick as their other dimensions are come across. If strength computes with classical strength formulas in analysis of these members, results could not display the accurate values, tension and skew bending formulas do not cover the whole procedure.

Application of loads to static system eccentrically causes torsion effect. The main difference appears with considering the torsion effect. Cross sections lose their linear format and cross section warping occurs based on the deformation which occurs under the torsion effect.

Torsion effect and cross section warping cause normal stress at a negligible level and shearing stress at thin walled members.

This present study includes a computer program which is developed to analyze thin walled continuous beams according torsion with warping, depending on Prof. A. Zafer ÖZTÜRK's associate professorship thesis. Required information about torsion with warping is presented in the initial sections of this study according to the thesis mentioned above.

**Keywords:** Thin walled steel, torsion with warping, transformation matrix method.



## 1. GİRİŞ

Konstrüktif elemanlar içerisinde ince cidarlı çubuklar iki şekilde tarif edilebilir:

- Cidar kalınlıkları, karakteristik ölçülerine (genişlik, yükseklik) göre küçük olan ve dolayısıyla dolu gövdeli çubuk olarak değerlendirilemeyecek çubuklar
- Kesit ölçüleri, uzunluğuna nazaran küçük olan prizmatik, silindirik veya hafif eğimli translasyon kabuklar.

Taşıyıcı sistem parçalarında bulunan enine yükler genellikle çubuk eksenine göre eksantrik etkiler ve böylece çubukta eğilmenin yanı sıra dönmeye zorlayan ilave tesirler meydana gelir.

20. yy başlangıcına kadar burulma hesabına, Barré de Saint-Venant tarafından geliştirilen ve 1855 de açıklanan teori esas teşkil etmiştir. Ancak kesitlerin düzlem kalması prensibinin (Bernoulli Kabulü) korunması, burulmanın teorik olarak açıklığa kavuşmasını uzun zaman önlemiştir. Bu kabulden ilk ayrılan St. Vénant teorisinde, çubukta ilave çarpılmalar meydana gelmektedir. Ancak bunun, taşıma kapasitesine etkisi ihmal edilmiştir. Fakat ihmal edilen bu kesit çarpılması boyuna yönde önemli gerilmeler meydana getirmektedir. Yapılan araştırmalar sonucunda çarpımalı burulma teorisi ortaya konulmuştur.

Çarpılma tabiri ile, düzlem bir kesitin noktalarının çubuğun boyuna doğrultusunda, kesit düzleminden dışarıya doğru yerdeğiştirmesinden meydana gelen deformasyonlar kastedilmektedir. Çubuk eksenini boyunca bir  $M_t$  burulma momenti değişiminden veya bir uç ankastreliğinden dolayı kesit serbestçe çarpılamıyorsa, burulma halinde bir zorlama oluyor demektir. Bu durum, bir çarpılma kuvvet grubuyla yani; çubukta çarpılma meydana getiren veya mevcut çarpılmayı değiştiren bileşkesiz ve bileşke momentsiz boyuna gerilmeler sistemi ile açıklanabilir.

Bu çalışmanın ikinci bölümünde genel tanımlar verilmiş; üçüncü ve dördüncü bölümlerde açık ve kapalı kesitlerin çarpımalı burulmaya göre hesap yönteminden bahsedilmiştir. Beşinci bölümde ise geçiş matrisleri metodunun esasları anlatılarak geçiş matrisi, başlangıç parametreleri, yük ve süreksizlik vektörlerinin çıkarılışından bahsedilmiştir. Buraya kadar olan kısımlar esas itibarıyla, özet bölümünde sözü edilen tezden aktarılmıştır.

Altıncı bölümde ise; ince cidarlı mütemadi kirişlerin çarpımalı burulma teorisine göre analizini gerçekleştirmek üzere geliştirilen bilgisayar programının (TWS) kullanılışı anlatılmış ve ayrıca akış diyagramları verilmiştir.

Ek.1'de TWS'nin analiz sonuçları ile ilgili sayısal örnekler, Ek.2'de de TWS'nin analizini

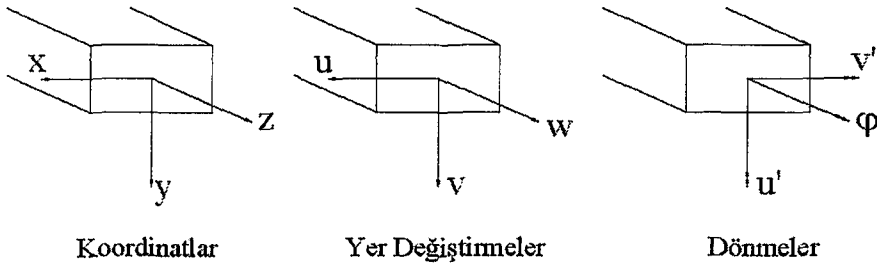
gerçekleştiren visual basic kodları verilmiştir.



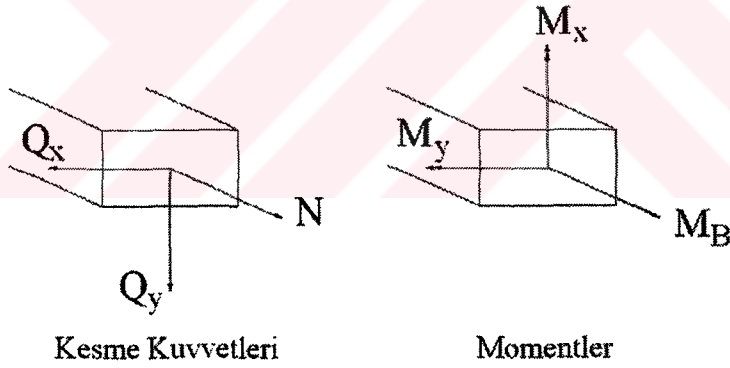
## 2. TANIMLAR

### 2.1 Koordinat Sistemi Ve Dış Tesirler

Koordinat sistemi olarak, z çubuk boylama ekseni olmak üzere bir x, y, z sağ koordinat sistemi seçilmiştir. u, v, w deformasyonları, ait oldukları eksenler doğrultusunda pozitif kabul edilmektedir. Pozitif dönmeler olarak, türevin-matematik tarifine göre  $u'=du/dz$  ve  $v'=dv/dz$  anlaşılacaktır. Pozitif kuvvetler ve momentler şekillerde gösterilmiştir.

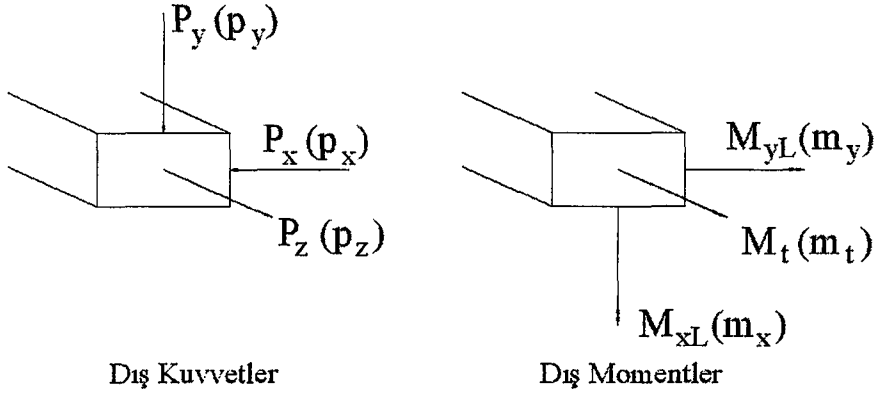


Şekil 2.1 – Koordinatlar sistemi (Öztürk, 1979)

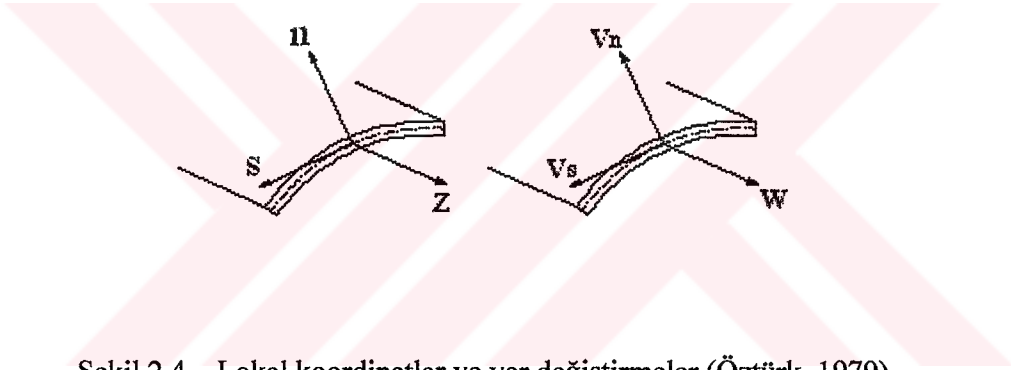


Şekil 2.2 – Kesit tesirleri (Öztürk, 1979)

İnce cidarlı çubuklarda xyz sistemi dışında bir de lokal koordinat sistemi göz önüne alınır (Şekil 2.4).

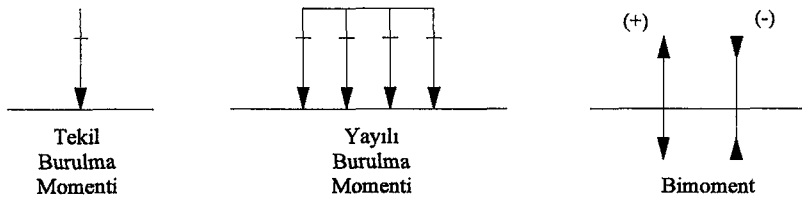


Şekil 2.3 – Dış tesirler (Öztürk, 1979)



Şekil 2.4 – Lokal koordinatlar ve yer değiştirmeler (Öztürk, 1979)

Burulma momentleri ve bimoment için şekil 2.5'te verilen gösteriş biçimi kullanılacaktır.



Şekil 2.5 – Burulma momenti ve bimoment pozitif notasyonu (Öztürk, 1979)

### 3. İNCE CİDARLI AÇIK KESİTLER

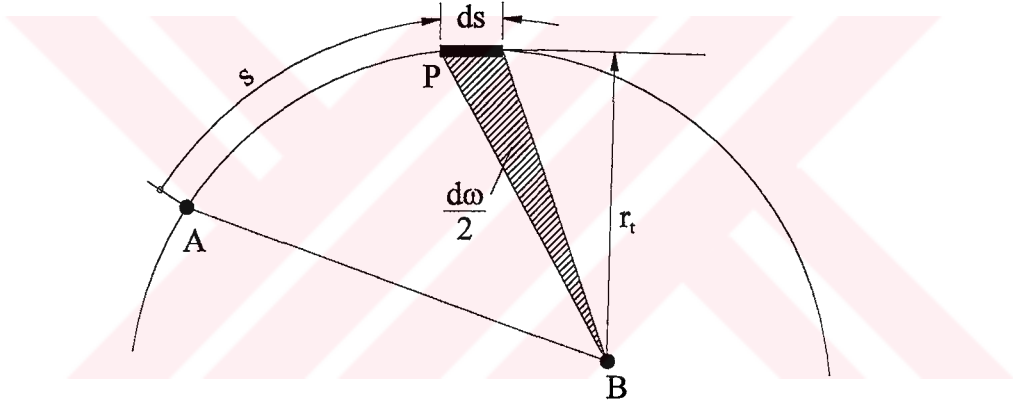
#### 3.1 Kabuller

İnce cidarlı açık kesilerin deformasyonları üzerinde şu kabuller yapılmaktadır.

- Deformasyonları sırasında kesit formu değişmiyor.
- Çubuk orta yüzeyindeki kayma deformasyonları ihmal ediliyor.
- Deformasyondan önce çubuk orta yüzeyine dik olan çizgisel bir eleman deformasyondan sonra da, doğrusal ve deforme olmuş orta yüzeye dik kalıyor.

#### 3.2 Kesit Değerlerinin Hesabı

##### 3.2.1 İnce Cidarlı Açık Kesitlerde $\omega$ Birim Çarpılma



Şekil 3.1 -  $\omega$  Birim çarpılma (Gülek, 1985)

Enkesit ortalama çizgisinin üzerinde bir A başlangıç noktası ile dışında bir B kıyaslama noktası seçilir (Şekil 3.1). Ortalama çizgi üzerinde bir P noktasının s eğrisel koordinatı, B ye göre saat hareketi yönünde (+) olarak alınır. P noktasında alınan ds elemanına karşın birim çarpılma diferansiyeli

$$d\omega = r_t \cdot ds \quad (3.1)$$

olarak tanımlanır. Burada  $r_t$ , B noktasından P deki teğete indirilen dikmenin boyudur. Buna göre tabanı ds, tepe noktası B olan üçgenin alanı

$$\frac{r_t \cdot ds}{2} = \frac{d\omega}{2} \quad (3.2)$$

dir. P noktasındaki birim çarpılma

$$\omega = \int_0^s r_t \cdot ds \quad (\text{cm}^2) \quad (3.3)$$

olarak tanımlanır. AP eğrisi ile BA ve BP ışınları arasında kalan alan  $w/2$  ye eşittir.

### 3.2.2 $J_o$ Burulma Mukavemeti

Burulma mukavemeti, membran analogisine göre burulma fonksiyonunun hacmi olarak hesaplanır. Buna göre ince dikdörtgen şeklindeki bir kesit için burulma mukavemeti

$$J_o = \frac{1}{3} \cdot a \cdot t^3 \quad (3.4)$$

bağıntısı ile hesaplanır. Eğer kesit dikdörtgenlerden meydana geliyorsa (3.4) yerine

$$J_o = \sum_i \frac{1}{3} \cdot a_i \cdot t_i^3 \quad (3.5)$$

yazılır.

### 3.2.3 Başlangıç Koordinatlarının Standartlaştırılması

Başlangıç koordinatlarının standartlaştırılması iki adımda yapılır. Birinci adımda eksen takımı ağırlık merkezine kaydırılır; ikinci adımda asal eksenler bulunur.

#### 1. Adım:

Oxy eksen takımı S ağırlık merkezine kaydırılır. Bu durumda

$$F_x = F_y = 0$$

olmalıdır. Ayrıca burulmada hiç bir bileşke normal kuvvet meydana gelmemesi şartından

$$F_\omega = 0$$

sağlanmalıdır. İlk değerler  $\hat{x}$ ,  $\hat{y}$  ve  $\hat{\omega}$  ile, ağırlık merkezine taşınan eksenler  $\bar{x}$ ,  $\bar{y}$  ve  $\bar{\omega}$  ile gösterilecektir. Bu değerler

$$\begin{aligned} \bar{x} &= \hat{x} - x_o \\ \bar{y} &= \hat{y} - y_o \\ \bar{\omega} &= \hat{\omega} - \omega_o \end{aligned} \quad (3.6)$$

formüllerini ile hesaplanır. Burada

$$\begin{aligned} x_o &= \frac{F_{\hat{x}}}{A} \\ y_o &= \frac{F_{\hat{y}}}{A} \\ \omega_o &= \frac{F_{\hat{\phi}}}{A} \end{aligned} \quad (3.7)$$

ile hesaplanır.

Kayma merkezine göre birim çarpılma koordinatı

$$\bar{\omega}_D = \bar{\omega} + \hat{y}_d \cdot \bar{x} + \hat{x}_d \cdot \bar{y} \quad (3.8)$$

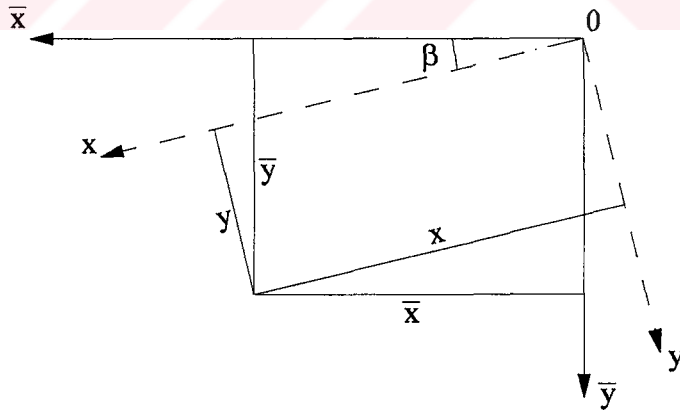
ile elde edilir.

## **2. Adım: Asal eksenlerin bulunması:**

Asal eksenlerin bulunması için ikinci mertebeden karışık üç integralin sıfırlanması gerekir:

$$F_{xy} = F_{\omega x} = F_{\omega y} = 0 \quad (3.9)$$

x, y asal eğilme eksenleri, ağırlık merkezine kaydırılmış eksen takımının (3.9) şartına uyuncaya kadar döndürülmesi ile elde edilir (Şekil 3.2).



Şekil 3.2 – x,y asal eksenleri (Öztürk, 1979)

Şekil 3.2'ye göre

$$\begin{aligned} x &= \bar{x} \cdot \cos \beta + \bar{y} \cdot \sin \beta \\ y &= -\bar{x} \cdot \sin \beta + \bar{y} \cdot \cos \beta \end{aligned} \quad (3.10)$$

yazılabilir.  $F_{xy}=0$  şartından aşağıdaki formüller elde edilir:

$$\operatorname{tg}2\beta = -\frac{F_{xy}}{F_{yy} - F_{xx}} \quad (3.11)$$

$$F_{xx} = \frac{1}{2} \cdot (F_{xx} + F_{yy}) - \frac{1}{2} \cdot \sqrt{(F_{yy} - F_{xx})^2 + 4 \cdot F_{xy}^2}$$

$$F_{yy} = \frac{1}{2} \cdot (F_{xx} + F_{yy}) + \frac{1}{2} \cdot \sqrt{(F_{yy} - F_{xx})^2 + 4 \cdot F_{xy}^2} \quad (3.12)$$

$F_{xx}$  ve  $F_{yy}$  kesitin en küçük ve en büyük atalet momentleri (asal atalet momentleri) dir.

$\omega$  asal birim çarpılma koordinatı için D kayma merkezine ait  $\bar{\omega}_D$  birim çarpılmasına (3.8) benzer olarak:

$$\omega(s) = \bar{\omega}(s) + d_y \cdot x - d_x \cdot y \quad (3.13)$$

yazılabilir. Burada  $d_x$  ve  $d_y$ , D kayma merkezinin, ilk dönme merkezi olarak alınan noktadan (ilk koordinat merkezi) x ve y doğrultularındaki uzaklığıdır. Bu değerler (3.9) denklemlerinden elde edilir:

$$d_y = -\frac{F_{\omega x}}{F_{xx}} \quad (3.14)$$

$$d_x = \frac{F_{\omega y}}{F_{yy}} \quad (3.15)$$

n sayıda ince dikdörtgen alanlardan oluşan bir kesitte yukarıda bahsi geçen değerler ( $F_{xy}$ ,  $F_{\omega x}$ ,  $F_{\omega y}$ ) aşağıdaki bağıntılarla hesaplanır.

$$A = \sum_{i=1}^n A_i \quad (3.16)$$

$$F_x = \frac{1}{2} \sum_{i=1}^n (x_a + x_b)_i \cdot A_i \quad (3.17)$$

$$F_y = \frac{1}{2} \sum_{i=1}^n (y_a + y_b)_i \cdot A_i \quad (3.18)$$

$$F_{\omega} = \frac{1}{2} \sum_{i=1}^n (\omega_a + \omega_b)_i \cdot A_i \quad (3.19)$$

$$F_{xx} = \frac{1}{3} \sum_{i=1}^n (x_a^2 + x_b^2 + x_a \cdot x_b)_i \cdot A_i \quad (3.20)$$

$$F_{yy} = \frac{1}{3} \sum_{i=1}^n (y_a^2 + y_b^2 + y_a \cdot y_b)_i \cdot A_i \quad (3.21)$$

$$F_{\omega\omega} = \frac{1}{3} \sum_{i=1}^n (\omega_a^2 + \omega_b^2 + \omega_a \cdot \omega_b)_i \cdot A_i \quad (3.22)$$

$$F_{xy} = \frac{1}{6} \sum_{i=1}^n (2 \cdot x_a \cdot y_a + 2 \cdot x_b \cdot y_b + x_a \cdot y_b + x_b \cdot y_a)_i \cdot A_i \quad (3.23)$$

$$F_{x\omega} = \frac{1}{6} \sum_{i=1}^n (2 \cdot x_a \cdot \omega_a + 2 \cdot x_b \cdot \omega_b + x_a \cdot \omega_b + x_b \cdot \omega_a)_i \cdot A_i \quad (3.24)$$

$$F_{y\omega} = \frac{1}{6} \sum_{i=1}^n (2 \cdot y_a \cdot \omega_a + 2 \cdot y_b \cdot \omega_b + y_a \cdot \omega_b + y_b \cdot \omega_a)_i \cdot A_i \quad (3.25)$$

### 3.3 Kesit Tesirleri

İnce cidarlı açık kesitlerde St. Vénant burulmasından meydana gelen primer burulma momenti

$$M_{BP} = G \cdot J_b \cdot \varphi' \quad (3.26)$$

izafi dönme açısı ile hesaplanır. Kesit çarpılmasından meydana gelen sekonder burulma momenti

$$M_{BS} = -E \cdot F_{\omega\omega} \cdot \varphi'''(z) \quad (3.27)$$

kesit dönme açısının üçüncü türevi ile hesaplanır. Kesit çarpılmasından oluşan bimoment ise

$$B = -E \cdot F_{\omega\omega} \cdot \varphi''(z) \quad (3.28)$$

kesit dönme açısının ikici türevi ile bulunmaktadır. Kesit tesirlerinin meydana getirdikleri normal ve kayma gerilmelerinin ifadeleri aşağıda verilmiştir.

Normal gerilmenin ifadesi

$$\sigma_z = \frac{B}{F_{\omega\omega}} \cdot \omega \quad (3.29)$$

şeklindedir. Primer burulma momentinden kaynaklanan kayma gerilmesi

$$\tau_p = \frac{M_{BP}}{J_b} \cdot \frac{\psi}{t} \quad (3.30)$$

Sekonder burulma momentinden meydana gelen kayma gerilmesi ifadesi de

$$\tau_{sk} = -\frac{M_{BS}}{F_{\omega\omega}} \cdot \frac{F_{\omega}(s)}{t(s)} \quad (3.31)$$

şeklindedir.

### 3.4 Çarpımlı Burulma Diferansiyel Denkleminin Çözümü

Çarpımlı burulma diferansiyel denklemi aşağıdaki gibidir:

$$E \cdot F_{\omega\omega} \cdot \varphi^{iv} - G \cdot J_b \cdot \varphi'' = m_t \quad (3.32)$$

bu denkleme

$$k^2 = \frac{G \cdot J_b}{E \cdot F_{\omega\omega}} \quad (3.33)$$

kısaltmasının yapılması ile

$$\varphi^{iv} - k^2 \cdot \varphi'' = \frac{m_t}{E \cdot F_{\omega\omega}} \quad (3.34)$$

şeklini alan diferansiyel denklemin çözümü, homojen  $\varphi_h$  ve özel  $\varphi_p$  çözüm kısımlarından oluşmaktadır:

$$\varphi = \varphi_h + \varphi_p \quad (3.35)$$

• Homojen Çözüm:

Hiperbolik fonksiyonlarla

$$\varphi_h = A_1 \cdot \sinh(k \cdot z) + A_2 \cdot \cosh(k \cdot z) + A_3 \cdot z + A_4 \quad (3.36)$$

yazılır.  $A_1 - A_4$ , sınır şartlarından belirlenecek katsayılardır.

(3.34) ve (3.36) nın çözümüyle, bimoment ve toplam burulma momenti ifadeleri ve  $\varphi$ ,  $\varphi'$  değerleri toplu halde şu şekilde gösterilirler:




$$\left. \begin{aligned}
 \varphi(z) &= A_1 \cdot \sinh(k \cdot z) + A_2 \cdot \cosh(k \cdot z) + A_3 \cdot z + A_4 + \varphi_p(z) \\
 \varphi'(z) &= A_1 \cdot k \cdot \cosh(k \cdot z) + A_2 \cdot k \cdot \sinh(k \cdot z) + A_3 + \varphi'_p(z) \\
 B(z) &= -E \cdot F_{\omega\omega} \cdot \varphi''(z) = -G \cdot J_b \cdot \left[ A_1 \cdot \sinh(k \cdot z) + A_2 \cdot \cosh(k \cdot z) + \frac{1}{k^2} \cdot \varphi''_p(z) \right] \\
 M_B(z) &= -E \cdot F_{\omega\omega} \cdot \varphi'''(z) + G \cdot J_b \cdot \varphi'(z) = G \cdot J_b \cdot \left[ A_3 + \varphi'_p(z) - \frac{1}{k^2} \cdot \varphi'''_p(z) \right]
 \end{aligned} \right\} (3.37)$$

• Özel Çözüm:

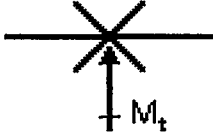

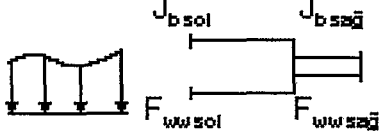
Değişik yüklemeler için ayrı ayrı bulunmalıdır. Özel çözüm sistemin çözümü ile elde edilecektir.

### 3.5 Sınır ve Geçiş Şartları

Çizelge 3.1 – Sınır Şartları (Öztürk, 1979)

Serbest Uç	Çatal Mesnet	Ankastre Mesnet
		
$\sigma_z = 0 \rightarrow B=0$	$\varphi=0$	$\varphi=0$
$M_B=0 \rightarrow GJ_b \varphi' = EF\omega\omega \varphi'''$	$\sigma_z = 0 \rightarrow B=0$	$\varphi'=0$

Çizelge 3.2 – Geçiş Şartları (Öztürk, 1979)

		
$\varphi=0$	$\varphi_{sol} = \varphi_{sağ}$	$\varphi_{sol} = \varphi_{sağ}$
$\varphi'_{sol} = \varphi'_{sağ}$	$\varphi'_{sol} = \varphi'_{sağ}$	$\varphi'_{sol} = \varphi'_{sağ}$
$B_{sol} = B_{sağ}$	$B_{sol} = B_{sağ}$	$B_{sol} = B_{sağ}$
$M_{Bsol} = M_{Bsağ} - M_t$	$M_{Bsol} = M_{Bsağ} + M_t$	$M_{Bsol} = M_{Bsağ}$
$M_{BPsol} = M_{BPsağ} - M_t$	$M_{BPsol} = M_{BPsağ} + M_t$	

### 3.6 Çubuk Hesabı

Çubuk üzerinde herhangi bir dış yük bulunmaması halinde, diferansiyel denklemin özel çözümü sıfır olur ve

$$\varphi(z) = \varphi_h(z) = A_1 \cdot \sinh(k \cdot z) + A_2 \cdot \cosh(k \cdot z) + A_3 \cdot z + A_4 \quad (3.38)$$

yazılır. Çözüm ifadesini daha elverişli hale getirmek için  $z=0$  kesitindeki bütün geometrik ve statik parametrelerin bilindiği kabul edilsin. Bu büyüklüklere Başlangıç Parametreleri denilir ve ( $\circ$ ) indisi ile gösterilecektir:

$z=0$  için

$$\left. \begin{aligned} \varphi_\circ &= A_2 + A_4 \\ \varphi'_\circ &= A_1 \cdot k + A_3 \\ B_\circ &= -A_2 \cdot G \cdot J_b \\ M_{B_\circ} &= A_3 \cdot G \cdot J_b \end{aligned} \right\} \quad (3.39)$$

(3.39) nın çözümünüyle katsayılar elde edilir:

$$\left. \begin{aligned} A_1 &= \frac{1}{k} \left( \varphi'_\circ - \frac{M_{B_\circ}}{G \cdot J_b} \right) \\ A_2 &= -\frac{B_\circ}{G \cdot J_b} \\ A_3 &= \frac{M_{B_\circ}}{G \cdot J_b} \\ A_4 &= \varphi_\circ + \frac{B_\circ}{G \cdot J_b} \end{aligned} \right\} \quad (3.40)$$

Bu değerlerle (3.37) yeniden yazılırsa:

$$\begin{aligned} \varphi(z) &= \varphi_\circ + \frac{1}{k} \cdot \varphi'_\circ \cdot \sinh(k \cdot z) + \frac{B_\circ}{G \cdot J_b} \cdot (1 - \cosh(k \cdot z)) + \frac{M_{B_\circ}}{G \cdot J_b} \cdot \left( z - \frac{1}{k} \cdot \sinh(k \cdot z) \right) \\ \varphi'(z) &= \varphi'_\circ \cdot \cosh(k \cdot z) - \frac{B_\circ}{G \cdot J_b} \cdot k \cdot \sinh(k \cdot z) + \frac{M_{B_\circ}}{G \cdot J_b} \cdot (1 - \cosh(k \cdot z)) \end{aligned} \quad (3.41)$$

$$\begin{aligned} B(z) &= -G \cdot J_b \cdot \varphi'_\circ \cdot \frac{1}{k} \cdot \sinh(k \cdot z) + B_\circ \cdot \cosh(k \cdot z) + M_{B_\circ} \cdot \frac{1}{k} \cdot \sinh(k \cdot z) \\ M_B(z) &= M_{B_\circ} \end{aligned}$$

Bu denklemler  $\varphi$ ,  $\varphi'$ ,  $B$  ve  $M_B$  nin hesabı için  $\varphi_\circ$ ,  $\varphi'_\circ$ ,  $B_\circ$  ve  $M_{B_\circ}$  başlangıç parametreleriyle lineer bir sistem oluştururlar. Aşağıda çizelge halinde gösterilen bu sisteme Başlangıç Parametreleri Matrisi denilmektedir (Çizelge 3.3).

Bu matris, başlangıç noktasının  $z=0$  yerine  $z=\xi$  ( $0<\xi<z$ ) çubuk kesitinin seçilmesi halinde, Çizelge 3.4' deki formu alır.

Çizelge 3.3 – Başlangıç parametreleri matrisi,  $z=0$  için (Öztürk, 1979)

	$\varphi_0$	$\varphi'_0$	$\frac{B_0}{G \cdot J_b}$	$\frac{M_{B_0}}{G \cdot J_b}$
$\varphi(z)$	1	$\frac{1}{k} \cdot \sinh kz$	$1 - \cosh kz$	$z - \frac{1}{k} \cdot \sinh kz$
$\varphi'(z)$	0	$\cosh kz$	$-k \cdot \sinh kz$	$1 - \cosh kz$
$\frac{B(z)}{G \cdot J_b}$	0	$-\frac{1}{k} \cdot \sinh kz$	$\cosh kz$	$\frac{1}{k} \cdot \sinh kz$
$\frac{M_B(z)}{G \cdot J_b}$	0	0	0	1

Çizelge 3.4 – Başlangıç parametreleri matrisi,  $z=\zeta$ ;  $0<\zeta<z$  için (Öztürk, 1979)

	$\varphi_\zeta$	$\varphi'_\zeta$	$\frac{B_\zeta}{G \cdot J_b}$	$\frac{M_{B_\zeta}}{G \cdot J_b}$
$\varphi(z)$	1	$\frac{1}{k} \cdot \sinh k(z-\zeta)$	$1 - \cosh k(z-\zeta)$	$(z-\zeta) - \frac{1}{k} \cdot \sinh k(z-\zeta)$
$\varphi'(z)$	0	$\cosh k(z-\zeta)$	$-k \cdot \sinh k(z-\zeta)$	$1 - \cosh k(z-\zeta)$
$\frac{B(z)}{G \cdot J_b}$	0	$-\frac{1}{k} \cdot \sinh k(z-\zeta)$	$\cosh k(z-\zeta)$	$\frac{1}{k} \cdot \sinh k(z-\zeta)$
$\frac{M_B(z)}{G \cdot J_b}$	0	0	0	1

Elde edilen bu çizelgeler Bölüm 5'de ince cidarlı çubukların geçiş matrisi metodu ile çözümünü sırasında kullanılacak olan geçiş matrisini oluşturacaktır.

## 4. İNCE CİDARLI KAPALI KESİTLER

### 4.1 Tek Gözlü Kapalı Kesitlerde Bredt Burulması

Kapalı kesitlerde burulma hesabı, St.Venant burulmasından faydalanılarak Bredt tarafından geliştirilmiştir.

$$\tau_t \cdot t = T = sb. \quad (4.1)$$

olarak ifade edilir. Burada T kayma akımının bütün kesitte sabit olduğu görülmektedir. Buradan hareketle

$$M_B = 2 \cdot T \cdot A_g \quad (4.2)$$

veya T ye göre çözerek

$$T = \frac{M_B}{2 \cdot A_g} \quad (4.3)$$

bulunur.  $\tau_t$  kayma gerilmesi

$$\tau_t = \frac{M_B}{2 \cdot A_g \cdot t} \quad (4.4)$$

ifadesi (Birinci Bredt Formülü) bulunur.

$\omega$  nın hesabı için, kayma gerilmesi ifadesinin s üzerinde entegre edilmesiyle

$$\omega(s) = \int_0^s \tau_t \cdot ds - \frac{T}{G \cdot \varphi'} \cdot \int_0^s \frac{1}{t} \cdot ds \quad (4.5)$$

bulunur. Buraya

$$\psi = \frac{T}{G \cdot \varphi'} \quad (4.6)$$

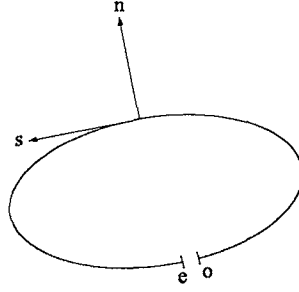
ile Birim Kayma Akımının yazılması sonucu birim çarpılma ifadesi elde edilir:

$$\omega(s) = \int_0^s \tau_t \cdot ds - \psi \int_0^s \frac{ds}{t} \quad (4.7)$$

Kapalı kesitlerde  $\omega$  nun başlangıç noktası belirli değildir. Bu sebeple, herhangi bir noktasından kesilerek açık hale getirilmiş kesitte,  $s$  üzerindeki entegrallerin başlangıçtan bitişe kadar tatbik edilmesi halinde  $\omega(0)=\omega_o$ ,  $\omega(e)=\omega_e$  ile

$$\omega_e = \omega_o$$

olmalıdır. (Şekil 4.1).



Şekil 4.1 – Kapalı kesitin kesilerek açık kesit haline getirilmesi (Öztürk, 1979)

Bu koşuldan hareketle

$$J_t = \psi \cdot 2 \cdot A_g \quad (4.8)$$

veya

$$J_t = \frac{4 \cdot A_g^2}{\oint \frac{ds}{t}} \quad (4.9)$$

kapalı kesitlerde ilave burulma mukavemeti terimi gelmektedir. Kesitin toplam burulma mukavemeti

$$J_b = J_o + J_t \quad (4.10)$$

olarak hesaplanır. Primer kayma gerilmesi ve sekonder kayma gerilmesi bağıntıları aşağıdaki gibi elde edilmiştir:

$$\tau_p = \frac{M_{BP}}{J_b} \cdot \frac{\psi}{t} \quad (4.11)$$

$$\tau_{sk}(s, z) = -\frac{M_{BS}}{F_{\omega\omega}} \cdot \frac{F_{\omega}(s)}{t} \quad (4.12)$$

Burada, kesilerek açık kesit haline getirilen tek gözlü kapalı kesitte  $s$  ye bağlı olarak  $F_{\omega_0}$  hesaplanacaktır. Açık hale getirilmesiyle meydana gelen farklılıktan dolayı  $F_{\omega_n}$  değeri ilave edilmelidir. Bu değer  $\omega_e = \omega_0$  süreklilik şartından hesaplanacaktır. Bu çözümün yapılmasıyla, tek gözlü kapalı kesitler için  $F_{\omega_n} = F_{\omega_1}$  olmak üzere

$$F_{\omega_1} = - \frac{\oint \frac{F_{\omega_0}(s)}{t} \cdot ds}{\oint \frac{ds}{t}} \quad (4.13)$$

bulunur.

#### 4.2 Çok Gözlü Kapalı Kesitlerde Bredt Burulması

Çok gözlü kesitler,  $M_B$  burulma momentinden dolayı aynı  $\phi'$  izafi dönmesine sahip tek gözlerin bir araya getirilmiş şekli olarak düşünülebilir. Bu durumda her münferit gözde,  $T_n$  kayma akımı ve  $\psi_n$  birim kayma akımı meydana gelir. (4.6) e göre:

$$\psi_n = \frac{T_n}{G \cdot \phi'_n} = \frac{T_n}{G \cdot \phi'} \quad (4.14)$$

$n$  inci gözdeki burulma momenti, (4.2) a benzer olarak

$$M_{Bn} = 2 \cdot T_n \cdot A_{gn} \quad (4.15)$$

yazılabilir. Kesitteki toplam moment

$$M_B = \sum M_{Bn} = 2 \cdot \sum T_n \cdot A_{gn} \quad (4.16)$$

ile bulunur.

Tek gözlü kapalı kesitlerdeki yöntemle benzer olarak, her bir göz, bir noktasından kesilerek açık kesit haline getirilir. Birim çarpılma koordinatlarında ( $\omega$ ) da süreklilik istendiği takdirde, yine

$$\omega_e - \omega_0 = 0$$

olmalıdır. Bu düşünce ile  $n$  inci göz için birim kayma akımları

$$-\Psi_{n-1} \int_{n-1,n} \frac{ds}{t} + \Psi_n \int_n \frac{ds}{t} - \Psi_{n+1} \int_{n,n+1} \frac{ds}{t} = 2 \cdot A_{gn} \quad (4.17)$$

şeklinde yazılabilir.

$\Psi_n$  in elde edilmesinden sonra diğer büyüklükler kolaylıkla bulunur:

(4.14), (4.15) ve (4.8) den:

$$J_t = 2 \sum_n \Psi_n \cdot A_{gn} \quad (4.18)$$

(4.11) ve (4.12)'ye göre

$$\tau_p = \frac{M_B}{J_b} \cdot \frac{\Psi_n}{t} \quad (4.19)$$

$$\tau_{sk}(s, z) = -\frac{M_{BS}}{F_{\omega\omega}} \cdot \frac{F_\omega(s)}{t} \quad (4.20)$$

bulunur. Çok gözlü kapalı kesitlerde, iki göz arasındaki cidarlar için iki  $F_{\omega n}$  değeri göz önüne alınmalıdır. Bu tip kesitler için bir denklem sistemi elde edilir. Bu denklem, n inci göz için aşağıdaki şekildedir:

$$-F_{\omega_{n-1}} \cdot \int_{n-1,n} \frac{ds}{t} + F_{\omega_n} \cdot \int_n \frac{ds}{t} - F_{\omega_{n+1}} \cdot \int_{n,n+1} \frac{ds}{t} = -\oint_n \frac{F_{\omega\omega}(s)}{t} \cdot ds \quad (4.21)$$

Bu denklem sistemi, Bredt burulmasında  $\psi$  kayma akımının hesabında olduğu gibi, yüklemenden bağımsız terimlere sahiptir.

$$F_\omega(s) = F_{\omega_0}(s) + F_{\omega_n} \quad (4.22)$$

alınması halinde, (4.20)'de kayma gerilmesinin hesabı için verilen -açık kesitlerdeki ifadenin aynısı- burada da geçerli olur.

## 5. İNCE CİDARLI MÜTEMADİ ÇUBUKLARIN HESABINDA GEÇİŞ MATRİSLERİ METODUNUN KULLANILMASI

Problemin kuruluşunda, sınır şartları yardımıyla belirtilmesi gereken değişken miktarının çok fazla olması, hesap açısından yorucu ve hata miktarını arttırıcı olacaktır. Bu sebepten değişken sayısını minimumda tutmak daha elverişlidir. Bunun sağlamak için esas fikir; sistemin tüm denklemlerini başlangıç bilinmeyenlerine göre kurarak; sistemi başlangıç değerleri problemine dönüştürmektir. Böylece ara süreksizlik şartlarından meydana gelecek yeni değişkenlerin önüne geçilmiş olunur. Bu yöntem hem daha sistematik bir çözüm yöntemi ortaya koyar, hem de programlama açısından daha kullanışlı bir methodur.

Teoride, değişkenin çeşitli değerleri arasında geçişi sağlayan bir matris esas rolü oynamaktadır. Bu bölümde, "Geçiş Matrisi" olarak ifade edilen bu matrisin ince cidarlı çubukların çözümü için oluşturulması ve uygulanması hakkında bilgi verilecektir.

Taşıyıcı sistemin herhangi bir  $i$  noktasında aranılan çarpılma büyüklükleri, bir  $v_i$  vektörü ile gösterilmektedir:

$$\begin{matrix} \text{Çubuk Dönmesi} \\ \text{İzafi Dönme(çarpılma oranı)} \\ \text{Bimoment} \\ \text{Toplam Burulma Momenti} \end{matrix} \begin{bmatrix} \varphi \\ \varphi' \\ B \\ M_B \end{bmatrix}^i = v_i \quad (5.1)$$

Bu büyüklükler, birbirleriyle matematik ilişkidedirler. Yer değiştirmeler ve kuvvetler arasında dördüncü dereceden bir diferansiyel ilişki mevcuttur. Bu diferansiyel denklem, birinci dereceden dört diferansiyel denklem sistemine dönüştürülebilir. Bu sistem,  $v_i$  vektörü yardımıyla

$$V' = A_d \cdot V + P \quad (5.2)$$

şeklinde gösterilebilir. Burada  $A_d$  diferansiyel matris ve  $P$  yük vektörüdür.

Bu diferansiyel denklemler sisteminin çözümü, homojen ve özel kısımlardan oluşmaktadır:

$$V(z) = V_H(z) + V_P(z) \quad (5.3)$$

Homojen çözüm,  $V_H(0)$  başlangıç vektörü  $z=0$  daki mevcut sınır değerlerini içerecek şekilde seçilirse, özel çözüm başlangıç noktasında sıfır olmalıdır. Böylece herhangi bir  $z$  kesitindeki bütün statik büyüklükler, her yükleme hali için başlangıç şartlarından hesaplanabilirler:

$$V(z) = VB(z) \cdot V(0) + VP(z) \quad (5.4)$$

Matris formundaki bu denklemde homojen kısım  $VB(z)$ , yükten bağımsız GEÇİŞ MATRİSİ ve özel kısım;  $VP(z)$ , ele alınan kiriş kesitinin yük vektörüdür. Bir taşıyıcı sistemin her noktasındaki geçiş matrisleri ve yük vektörleri bilinirse, statik büyüklükler bilinen başlangıç değerlerinden matris hesabı ile bulunabilir.

### 5.1 Geçiş Matrisinin Elde Edilmesi

Çubuk dönmesini temsil eden  $\varphi$  ile izafi dönmeyi gösteren  $\varphi'$  arasında, bilinen (3.9) bağıntısı mevcuttur:

$$\varphi' = \frac{\partial \varphi}{\partial z} \quad (5.5)$$

Bimoment ve toplam burulma momenti ile  $\varphi$  arasında ise

$$B = -E \cdot F_{\omega\omega} \cdot \varphi'' \quad (5.6)$$

$$M_B = -E \cdot F_{\omega\omega} \cdot \varphi''' + G \cdot J_b \cdot \varphi' \quad (5.7)$$

ilişkisinin olduğu bilinmektedir. (5.7) ifadesi (5.6) yardımıyla,

$$M_B = B' + G \cdot J_b \cdot \varphi' \quad (5.8)$$

veya

$$B' = -G \cdot J_b \cdot \varphi' + M_B \quad (5.9)$$

şeklinde yazılabilir. Ayrıca mevcut yüklemekten dolayı, burulma momentinin dengesinden

$$m_t + \frac{dM_B}{dz} = 0 \quad (5.10)$$

olacağı açıktır. (5.5), (5.6), (5.9) ve (5.10) ile  $V'(z)$  vektörü

$$\left. \begin{array}{l} \varphi' = \varphi' \\ \varphi'' = -\frac{1}{E \cdot F_{\omega\omega}} B \\ B' = -G \cdot J_b \cdot \varphi' + M_B \\ M'_B = -m_t \end{array} \right\} \quad (5.11)$$

ve böylece diferansiyel denklem sistemi, matris formunda

$$\begin{bmatrix} \varphi \\ \varphi' \\ B \\ M_B \end{bmatrix}' = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -\frac{1}{E \cdot F_{\omega\omega}} & 0 \\ 0 & -G \cdot J_b & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \varphi \\ \varphi' \\ B \\ M_B \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ -m_t \end{bmatrix} \quad (5.12)$$

$$V' = A_d \cdot V + P \quad (5.13)$$

şeklini alır.

$E \cdot F_{\omega\omega} = sb$  ve  $m_t = sb$  halinde bu diferansiyel denklemin çözümüyle, yüklemenden ve sınır şartlarından bağımsız VB geçiş matrisi ve yalnız yüklemeye bağlı VP yük vektörü aşağıdaki gibi elde edilirler:

$$VB = \begin{bmatrix} 1 & \frac{1}{k} \sinh kz & \frac{1 - \cosh kz}{G \cdot J_b} & \frac{1}{G \cdot J_b} \left( z - \frac{1}{k} \sinh kz \right) \\ 0 & \cosh kz & -\frac{k \cdot \sinh kz}{G \cdot J_b} & \frac{1 - \cosh kz}{G \cdot J_b} \\ 0 & -\frac{1}{k} \sinh kz & \frac{\cosh kz}{G \cdot J_b} & \frac{1}{G \cdot J_b} \frac{1}{k} \sinh kz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.14)$$

veya  $G \cdot J_b$  katsayısı  $V(z)$  ve  $V(0)$  vektörleri içine alınarak;

$$VB = \begin{bmatrix} 1 & \frac{1}{k} \sinh kz & 1 - \cosh kz & z - \frac{1}{k} \sinh kz \\ 0 & \cosh kz & -k \cdot \sinh kz & 1 - \cosh kz \\ 0 & -\frac{1}{k} \sinh kz & \cosh kz & \frac{1}{k} \sinh kz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.15)$$

$$VP = \begin{bmatrix} -m_t \left[ \frac{z^2}{2} + \frac{1}{k^2} (1 - \cosh kz) \right] \\ -m_t \left( z - \frac{1}{k} \sinh kz \right) \\ m_t \frac{1}{k^2} (1 - \cosh kz) \\ -m_t \cdot z \end{bmatrix} \quad (5.16)$$

(5.4) çözümündeki  $V(0)=VX$  başlangıç değerleri vektörü için bu durumda,  $GJ_b$  katsayısı dikkate alınarak

$$VX = \begin{bmatrix} G \cdot J_b \cdot \varphi_o \\ G \cdot J_b \cdot \varphi'_o \\ B_o \\ M_{B_o} \end{bmatrix} \quad (5.17)$$

yazılmalıdır. Geçiş matrisindeki  $GJ_b$  katsayısının dışarı alınmasıyla

$$V(z) = \begin{bmatrix} G \cdot J_b \cdot \varphi(z) \\ G \cdot J_b \cdot \varphi'(z) \\ B(z) \\ M_B(z) \end{bmatrix} \quad (5.18)$$

şeklini alan  $V(z)$  vektöründe, ikinci terimin  $M_{BP}$  ( St. Vénant burulma momenti) olduğu dikkat çekicidir.

Açık kesitli çubuk için bulunan (5.15) geçiş matrisi, kapalı kesitli çubuklarda,  $\rho$  katsayısı farkıyla aşağıdaki formu alır:

$$VB = \begin{bmatrix} 1 & \frac{1}{\rho k} \sinh kz & 1 - \cosh kz & z - \frac{1}{\rho k} \sinh kz \\ 0 & \frac{1}{\rho} \cosh kz & -k \cdot \sinh kz & 1 - \frac{1}{\rho} \cosh kz \\ 0 & -\frac{1}{\rho k} \sinh kz & \cosh kz & \frac{1}{\rho k} \sinh kz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.19)$$

Görüldüğü gibi, (5.19) de  $\rho=1$  olması halinde (5.15) elde edilir.

Kapalı kesitli ince cidarlı çubuklarda, sekonder kayma kuvvetlerinden gelen tesirleri de ihtiva eden  $\vartheta$  ( $GJ_b \vartheta$ ) çarpılma ölçüsünün (izafî dönmenin) bulunmak istenilmesi halinde  $V(z)$ ,  $VP(z)$  vektörlerine ve  $VB(z)$  matrisine bir beşinci satır ilave etmek gerekir.  $VX$  başlangıç vektöründe bir değişiklik olmayacaktır:

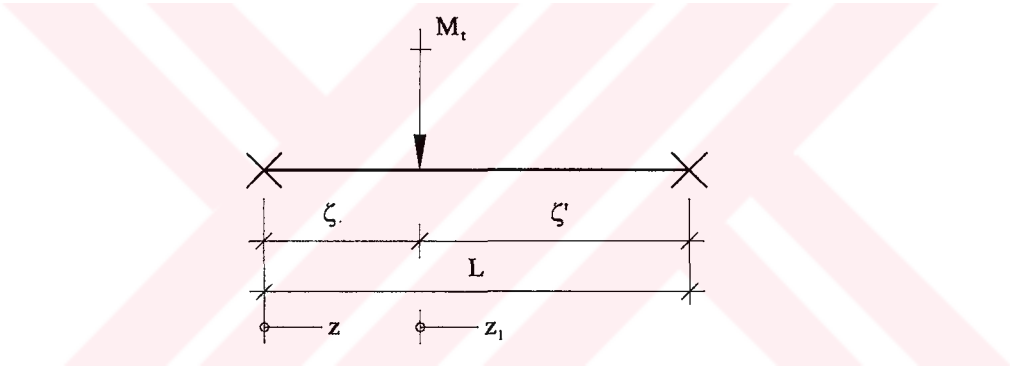
$$V(z) = \begin{bmatrix} G \cdot J_b \cdot \varphi(z) \\ G \cdot J_b \cdot \varphi'(z) \\ B(z) \\ M_B(z) \\ G \cdot J_b \cdot \vartheta(z) \end{bmatrix} \quad (5.20)$$

$$VB(z) = \begin{bmatrix} 1 & \frac{1}{\rho k} \sinh kz & 1 - \cosh kz & z - \frac{1}{\rho k} \sinh kz \\ 0 & \frac{1}{\rho} \cosh kz & -k \cdot \sinh kz & 1 - \frac{1}{\rho} \cosh kz \\ 0 & -\frac{1}{\rho k} \sinh kz & \cosh kz & \frac{1}{\rho k} \sinh kz \\ 0 & 0 & 0 & 1 \\ 0 & \cosh kz & -\rho k \cdot \sinh kz & 1 - \cosh kz \end{bmatrix} \quad (5.21)$$

## 5.2 Yük Vektörünün Elde Edilmesi

Dış yükleme durumları ve ara mesnetler çubukta bir süreksizlik meydana getirir. Bu süreksizliklerin çözüm sistemi içersine katılması gerekir. Aşağıda farklı durumlar için bu vektörlerin elde edilişi yer almaktadır.

### 5.2.1 Münferit Burulma Momenti



Şekil 5.1 – Münferit burulma momenti yüklemesi (Öztürk, 1979)

Münferit burulma momenti yüklemesi  $z=\zeta$  noktasında bir süreksizlik meydana getirir. Bu süreksizlik vektörü aşağıdaki gibi ifade edilir:

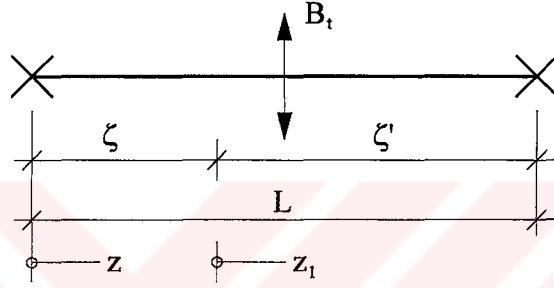
$$K(z_1) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -M_t \end{bmatrix} \quad (5.22)$$

Olur.  $z=L$  için (süreksizlik vektörü açıklık sonu için geçiş matrisi ile çarpımı sonucu) aşağıdaki formu alır.

$$VP(Z) = \begin{bmatrix} -M_t(z_1 - \frac{1}{\rho k} \sinh kz_1) \\ -M_t(1 - \frac{1}{\rho} \cosh kz_1) \\ -M_t \frac{1}{\rho k} \sinh kz_1 \\ -M_t \\ -M_t(1 - \cosh kz_1) \end{bmatrix} \quad (5.23)$$

Münferit burulma momenti yük matrisi elde edilmiş olur.

### 5.2.2 Münferit Bimoment



Şekil 5.2 – Münferit bimoment yüklemesi (Öztürk, 1979)

Münferit bimoment yüklemesi durumu için de süreksizlik vektörü aşağıdaki gibi yazılır:

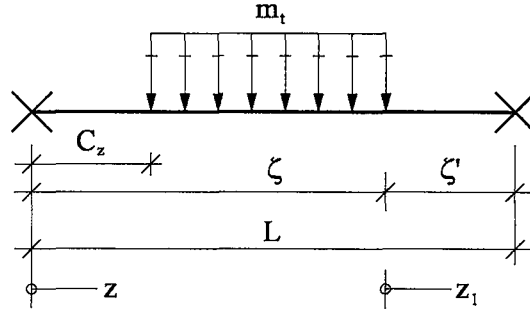
$$K(Z_1) = \begin{bmatrix} 0 \\ 0 \\ -B_t \\ 0 \end{bmatrix} \quad (5.24)$$

Geçiş matrisi ile çarpım işlemi yapıldıktan sonra;  $z=L$  için yük vektörü:

$$VP(Z) = \begin{bmatrix} -B_t(1 - \cosh kz_1) \\ B_t \cdot k \cdot \sinh kz_1 \\ -B_t \cosh kz_1 \\ 0 \\ B_t \cdot \rho k \cdot \sinh kz_1 \end{bmatrix} \quad (5.25)$$

olarak elde edilir.

### 5.2.3 Yayılı Burulma Momenti



Şekil 5.3 – Yayılı burulma momenti yüklemesi (Öztürk, 1979)

Yayıllı burulma momentinden meydana gelen süreksilik vektörü:

$$K(z) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -m_t(z) \end{bmatrix} \quad (5.26)$$

olur. Geçiş matrisi ile çarpım işlemi yapıldıktan sonra  $c_z < z < \zeta$  için yük vektörü elde edilmiş olur:

$$VP(Z) = \begin{bmatrix} -m_t \left[ \frac{(z-c_z)^2}{2} + \frac{1}{\rho k^2} (1 - \cosh k(z-c_z)) \right] \\ -m_t \left[ z - c_z - \frac{1}{\rho k} \sinh k(z-c_z) \right] \\ m_t \frac{1}{\rho k^2} [1 - \cosh k(z-c_z)] \\ -m_t (z - c_z) \\ -m_t \left[ z - c_z - \frac{1}{k} \sinh k(z-c_z) \right] \end{bmatrix} \quad (5.27)$$

### 5.3 Ara Mesnetler

İki veya daha fazla çubuktan oluşan sürekli kirişlerdeki ara mesnetler ilave bilinmeyenleri ortaya çıkarmaktadır. Çözümün yapılabilmesi için bilinmeyen adedi kadar bilinene ihtiyaç vardır.

Ara mesnetlerde bilinen dönmenin ( $\varphi=0$ ) sıfır olması, bilinmeyen ise  $M_B$  toplam burulma momentidir. İşleme girecek olan bilinmeyen vektörü:

$$K = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (5.28)$$

olur.

Bu şartlar ile mütemadi kirişlerin çözümünde süreklilik sağlanmış olur. Ara mesnetler üzerinde bilinenler vasıtasıyla denklemler elde dillecek ve bu denklem sisteminin çözümü ile başlangıç değerleri bulunacaktır. Daha sonra sistemin istenen herhangi bir noktasında çarpılma büyüklükleri elde edilebilir.



## 6. İNCE CİDARLI MÜTEMADİ ÇUBUKLARIN HESABININ BİLGİSAYAR PROGRAMI İLE YAPILMASI












### 6.1 TWS Programının Tanıtımı

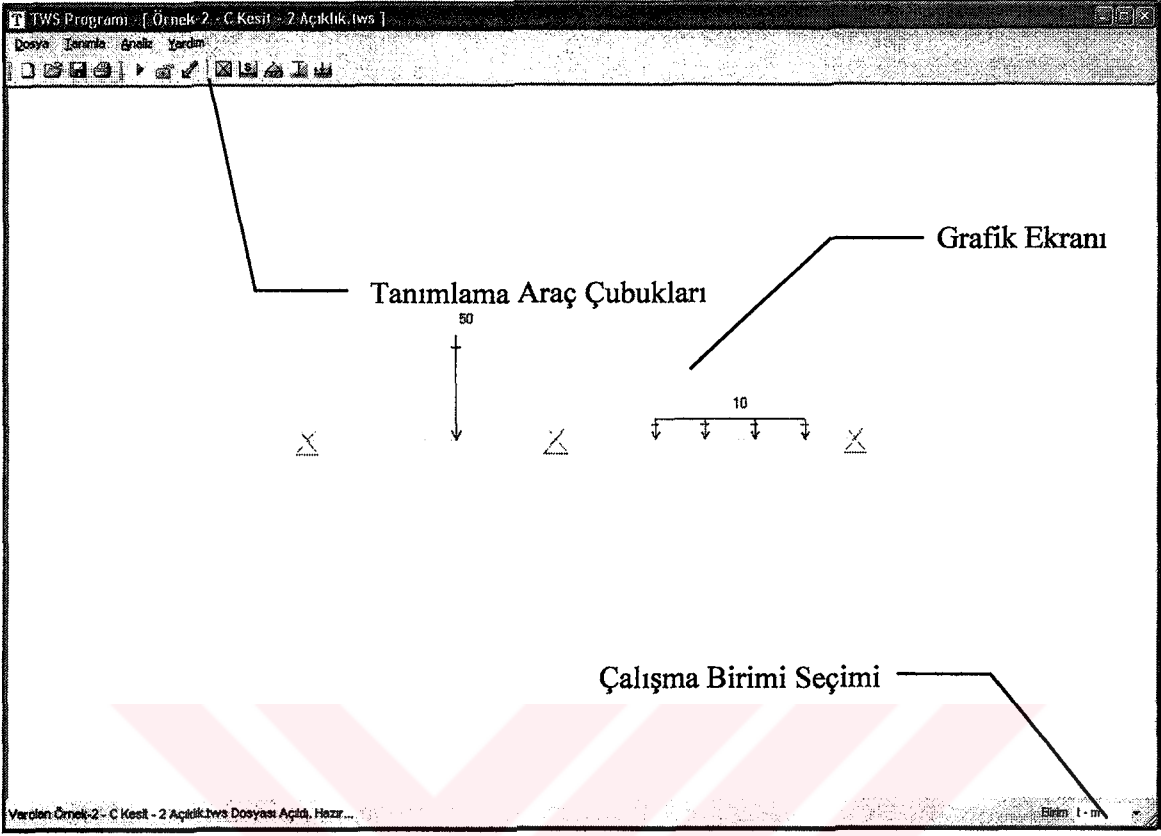
Bu çalışmada, ince cidarlı mütemadi çubukların çarpımalı burulma analizini gerçekleştirecek bir bilgisayar programı geliştirilmiştir. Bu programdan metin içerisinde TWS olarak bahsedilecektir.

TWS programı, sürekli çubukların analizini başlangıç değerleri metodu ve geçiş matrisi metodu ile gerçekleştirmektedir. TWS; I, C ve Z açık enkesitleri, tek ve iki gözlü kapalı enkesit olmak üzere beş farklı enkesit için çözüm yapmaktadır. TWS programı Visual Basic programlama dili kullanılarak yazılmıştır.

### 6.2 TWS Programıyla Sistem Tanımı ve Analizi

Şekil 6.1'de TWS programının ana penceresi görülmektedir. Ana pencerede tanımlanan sistemin görüntülediği bir grafik arayüz ekran, tanımlama menülerine erişimi sağlayan araç çubukları ve menü, çalışma biriminin seçilebildiği bir seçim kutusu yer almaktadır.

Araç çubuklarındaki butonlar sırası ile şu görevleri yapmaktadır:  Butonu analizi başlatır,  butonu tamamlanan analiz sonucunda kilitlenen tanımlama butonlarının kilidini kaldırır.  Ekrandaki görüntüyü tazeler.  butonu malzeme bilgileri tanımlama formunu açar.  butonu sistem tanımlama formunu açar.  butonu mesnetlenme şartları tanımlama formunu açar.  butonu enkesit bilgileri tanımlama formunu açar.  butonu yükleme bilgileri tanımlama formunu açar.  butonu analiz sonucunda moment diyagramlarını görüntüler.  butonu analiz sonucu raporunu görüntüler.  butonu enkesit gerilme diyagramlarını görüntüler.



Şekil 6.1 – TWS Programı Ana Penceresi

Şekil 6.2 – Malzeme Bilgileri Tanımlama Formu

Malzeme bilgileri tanımlama formunda elastisite modülü ve kayma modülü ana pencerede seçilen birim cinsinden tanımlanır.

	Uzunluk (L)
1. Açıklık	5
2. Açıklık	6

Şekil 6.3 – Sistem Bilgileri Tanımlama Formu

Sistem bilgileri tanımlama formunda sistem açıklık sayısı ve her açıklığın uzunluklarının tanımlamaları yapılır.

Mesnet Numarası	Mesnet Şartı
1. Mesnet	Çatal
2. Mesnet	Çatal
3. Mesnet	Çatal

Şekil 6.4 – Mesnet Şartları Tanımlama Formu

Mesnet şartları tanımlama formunda, sınır şartları çatal mesnet, ankastre mesnet veya serbest uç olarak tanımlanır.

**Kesit Bilgi Girişi**

Kesit Tipi Seçimi

Tamam  
İptal

Birim: t - m

Kesit Bilgileri

Üst Başlık Genişliği (b<sub>ust</sub>) | .3  
 Üst Başlık Kalınlığı (t<sub>ust</sub>) | .02  
 Gövde Yüksekliği (h) | .5  
 Gövde Kalınlığı (t<sub>gov</sub>) | .016  
 Alt Başlık Genişliği (b<sub>alt</sub>) | .3  
 Alt Başlık Kalınlığı (t<sub>alt</sub>) | .02  
 Üst Başlık Taşmaları (b1) |

Şekil 6.5 – Kesit Bilgi Giriş Formu

Bu menüde enkesit tipi ve boyut bilgileri tanımlamaları yapılır.

**Burulma Momenti ve Bimoment Yüklemesi**

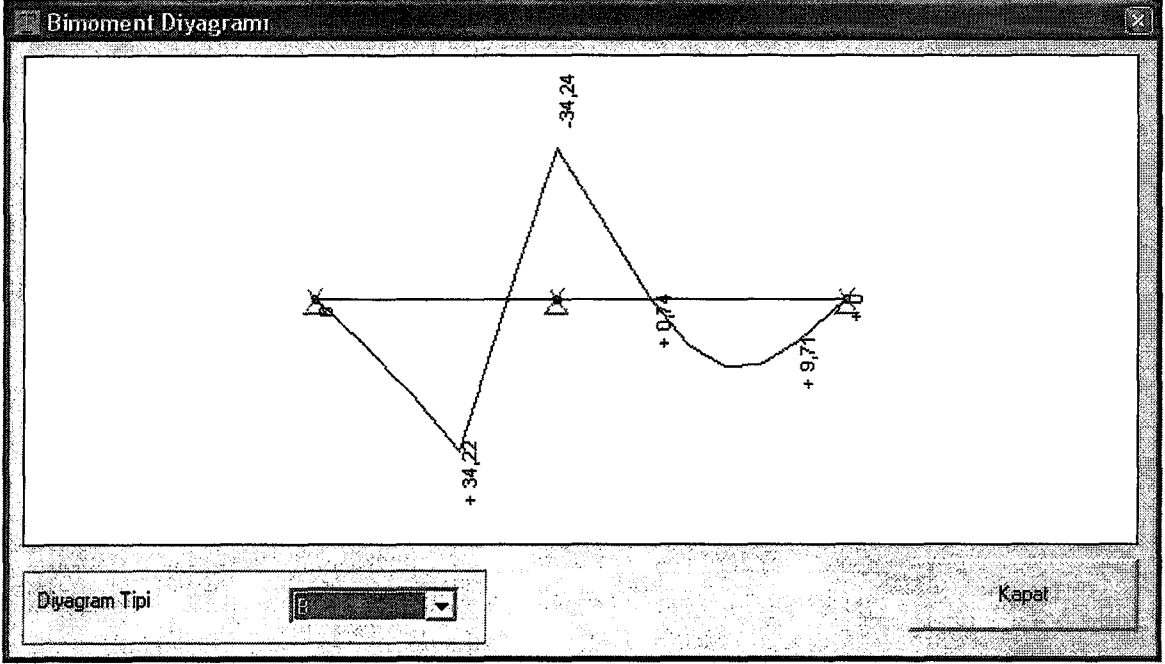
Açıklık No	Yükleme Tipi	Yükleme Değeri	Açıklık	Z <sub>a</sub> Mesafesi	Z <sub>c</sub> Mesafesi
1. Açıklık	Yüksüz	Tekil Burulma Mom.	50	3	0
2. Açıklık		Yayıllı Burulma Mom.	10	5	2

Birim: t - m


Tamam  
İptal

Şekil 6.7 – Yükleme Bilgileri Tanımlama Formu

Yükleme bilgileri tanımlama formunda tekil burulma momenti, yayıllı burulma momenti ve tekil bimoment yükelemeleri tanımlamaları yapılır.



Şekil 6.8 – Moment Diyagramları Görüntüleme Formu

Analiz sonucunda moment diyagramları,  butonu tıklanıldığı zaman yukarıdaki form vasıtasıyla görüntülenir. Diyagram tipi seçeneğinden Toplam Burulma Momenti (MB), Primer Burulma Momenti (MBP), Sekonder Burulma Momenti (MBS) ve Bimoment (B) diyagramları görüntülenebilir.

Şekil 6.9 – Analiz Sonuçları Seçenekleri Formu

Bu form sayesinde analiz sonucu raporunda görüntülenmesi istenilen bileşenler seçilebilmektedir. İsteğe bağlı olarak sistem üzerinde belirli noktalarındaki burulma momentleri ve  $\phi$  dönme açısı değerleri tablosu; sistem moment diyagramları ve belirlenen enkesit üzerinde oluşan, primer ve sekonder kayma gerilmeleri ve çekme gerilmeleri diyagramlarının rapor üzerinde görüntülenmesi sağlanır.

**TWS Programı - Dosya Adı : Örnek-2 - C Kesit - 2 Açıklık.tws** Sayfa : 1

**Sistem Bilgileri:**

Çalışma Birimi : t - m  
Açıklık Sayısı : 2

**Açıklık Mesafeleri:**

1. Açıklık : 5 m  
2. Açıklık : 6 m

**Mesnet Şartları:**

1. Mesnet : Çatal Mesnet  
2. Mesnet : Çatal Mesnet  
3. Mesnet : Çatal Mesnet

**Yükleme Bilgileri:**

Açıklık	Yükleme Tipi	Yükleme Değeri	Za	Cz
1. Açıklık	Tekil Burulma Momenti	50	t-m	3 m
2. Açıklık	Yayıllı Burulma Momenti	10	t-m	5 m

**Enkesit Bilgileri:**

Kesit Tipi : C Kesit  
Üst Başlık Geniřliđi (bust) : 0,3 m  
Üst Başlık Kalınlıđı (tust) : 0,02 m  
Gövde Yüksekliđi (h) : 0,5 m  
Gövde Kalınlıđı (tgv) : 0,016 m  
Alt Başlık Geniřliđi (balt) : 0,3 m  
Alt Başlık Kalınlıđı (talt) : 0,02 m

**Enkesit Deđerleri:**

Şekil 6.10 – Analiz Sonucu Raporu

Bu formda analiz sonuçları bir rapor halinde görüntülenmekte, hesap sonuçları incelenebilmektedir.

**Enkesit Gerilme Diyagramları**

Gerilme İstenen Açıklık : 1. Açıklık

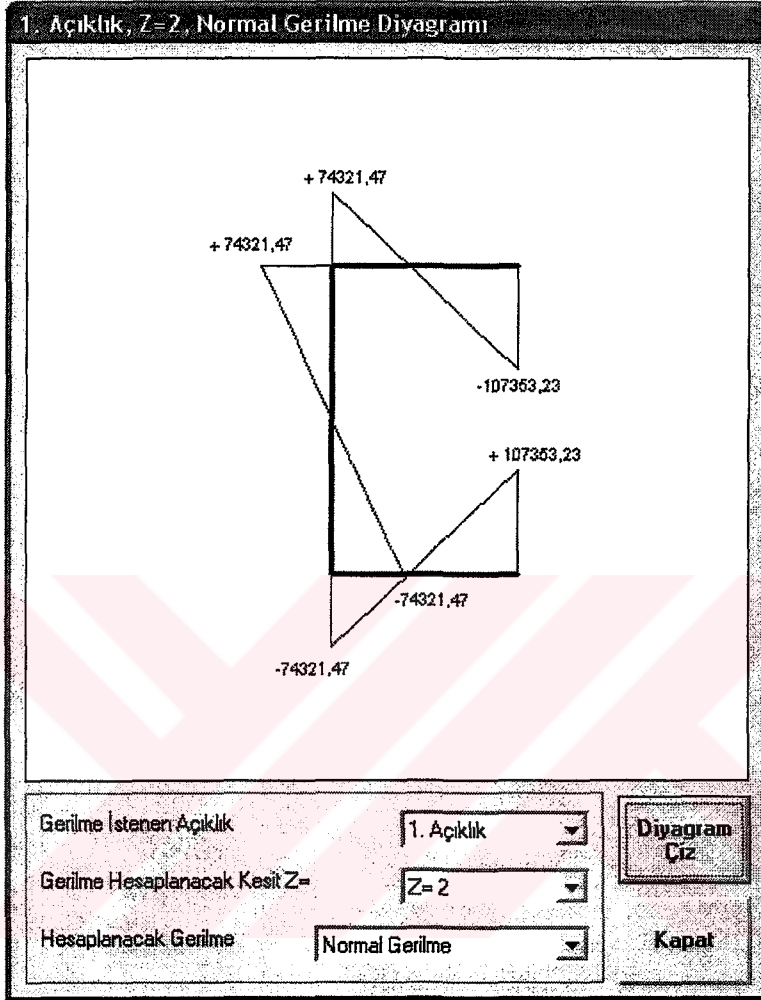
Gerilme Hesaplanacak Kesit Z= : Z=0

Hesaplanacak Gerilme : Normal Gerilme

Tamam

Şekil 6.11 – Enkesit gerilme diyagramları seçenek formu

Bu form ile görüntülenecek enkesit gerilme tipini, gerilme diyagramı çizilecek olan açıklık ve enkesit mesafesi seçenekleri düzenlenir.



Şekil 6.12 – Enkesit gerilme diyagramı formu

Bu formda enkesit gerilme diyagramları görüntülenir. Form üzerindeki seçeneklerden istenilen açıklık, gerilme hesaplanacak enkesit ve görüntülenmesi istenilen gerilme tipi seçilebilir.

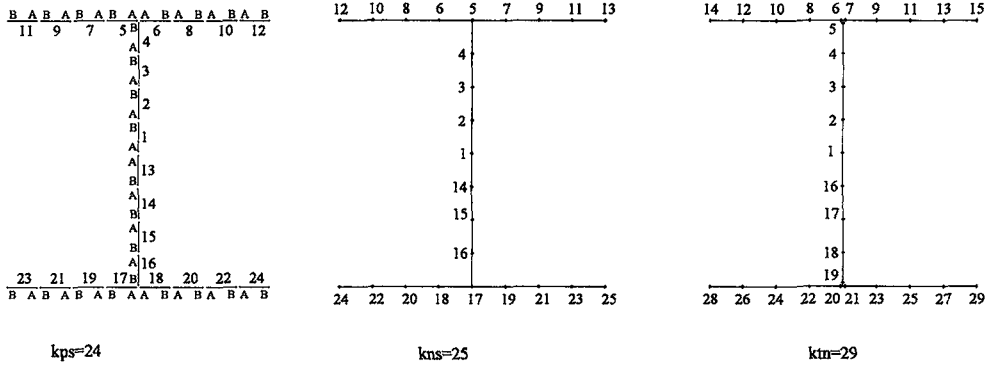
### 6.3 TWS Programının Çözüm Yöntemi

#### 6.3.1 TWS Programı Enkesit Tipleri

Programda beş tip enkesit kullanılmıştır. Bunlar I, C, Z açık kesitler ve tek gözlü ve çift gözlü kapalı kesitlerdir. Bu kesitlerin program hesabında kullanılması ve gerilme hesapları ile ilgili bilgiler aşağıda verilmiştir.

- I Kesit:

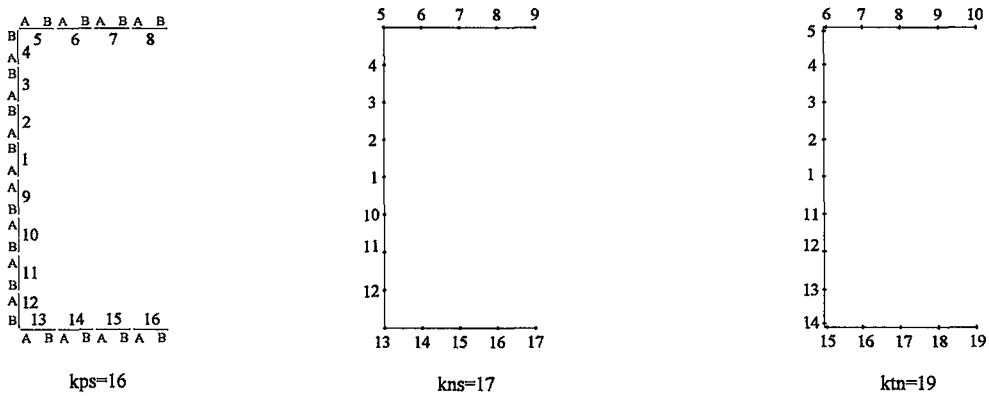
I kesit, 24 kesit parçasından (kps) oluşmaktadır. Kesit üzerinde normal gerilme hesaplanacak noktaların sayısı (kns) 25, kayma gerilmeleri hesaplanacak nokta sayısı (ktn) 29 dur.



Şekil 6.13 – I Kesit bilgileri

- C Kesit:

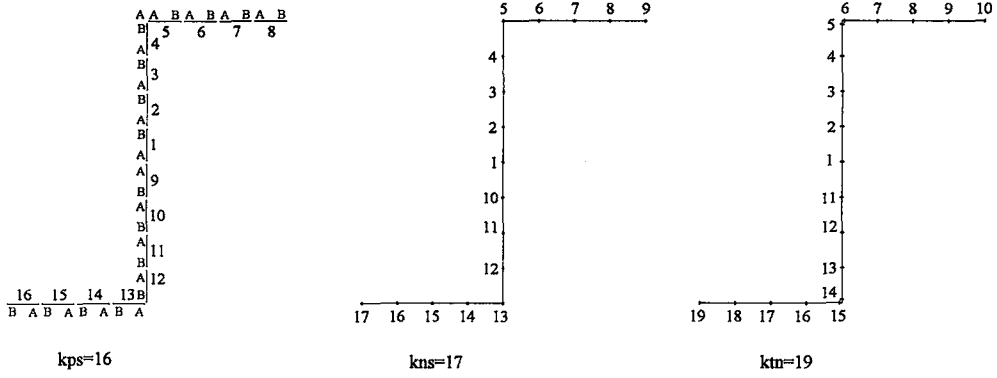
C kesit, 16 kesit parçasından (kps) oluşmaktadır. Kesit üzerinde normal gerilme hesaplanacak noktaların sayısı (kns) 17, kayma gerilmeleri hesaplanacak nokta sayısı (ktn) 19 dur.



Şekil 6.14 – C Kesit bilgileri

- Z Kesit:

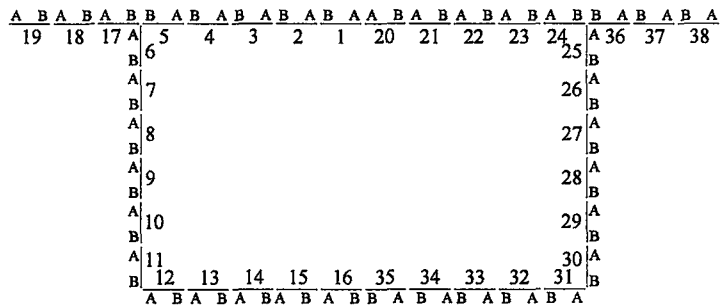
C kesit, 16 kesit parçasından (kps) oluşmaktadır. Kesit üzerinde normal gerilme hesaplanacak noktaların sayısı (kns) 17, kayma gerilmeleri hesaplanacak nokta sayısı (ktn) 19 dur.



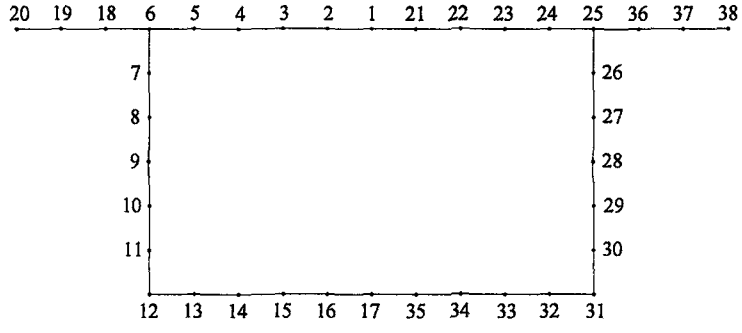
Şekil 6.15 – C Kesit bilgileri

- Tek Gözlü Kapalı Kesit:

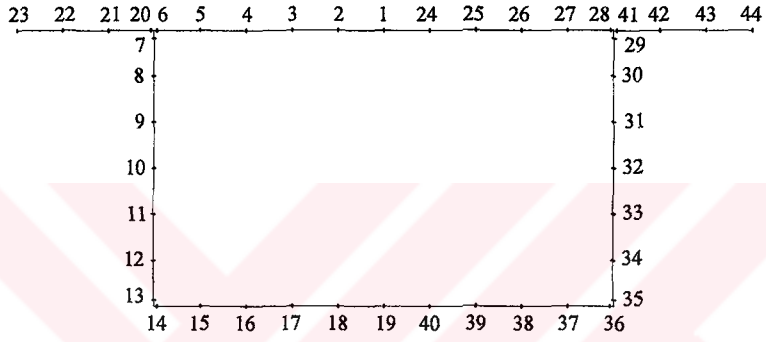
Tek gözlü kapalı kesit, 38 kesit parçasından (kps) oluşmaktadır. Kesit üzerinde normal gerilme hesaplanacak noktaların sayısı (kns) 38, kayma gerilmeleri hesaplanacak nokta sayısı (ktn) 44 dur.



kps=38



$kns=38$



$ktn=44$

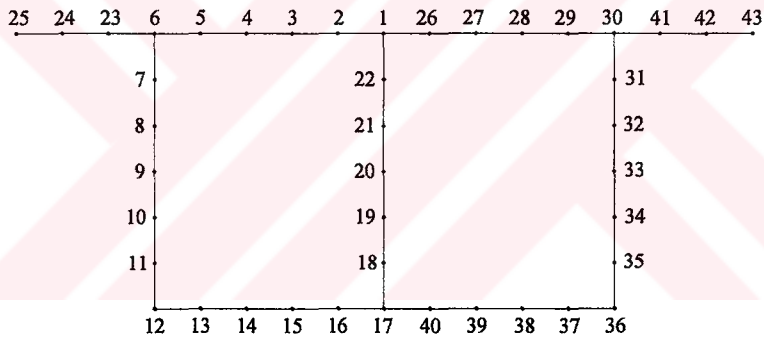
Şekil 6.16 – Tek gözlü kapalı kesit bilgileri

- Çift Gözlü Kapalı Kesit:

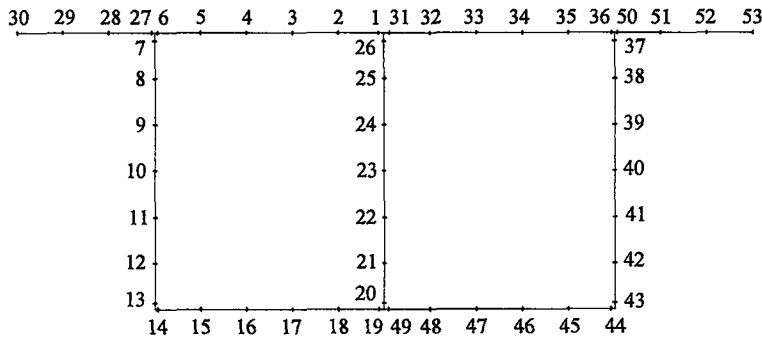
Çift gözlü kapalı kesit, 38 kesit parçasından (kps) oluşmaktadır. Kesit üzerinde normal gerilme hesaplanacak noktaların sayısı (kns) 38, kayma gerilmeleri hesaplanacak nokta sayısı (ktn) 44 dur.

A	B	A	B	A	B	B	A	B	A	B	A	B	A	B	A	A	B	A	B	A	B	A	B	A	B	B	A	B	A	B	A	B	A	B	A			
25	24	23	A	5	4	3	2	1	B	26	27	28	29	30	A	42	43	44																				
			B	6					A	22					B	31																						
			A	7					B	21					A	32																						
			B	8					A	20					B	33																						
			A	9					B	19					A	34																						
			B	10					A	18					B	35																						
			A	11					B	17					A	36																						
			B	12	13	14	15	16	A	41	40	39	38	37	B																							

kps=44



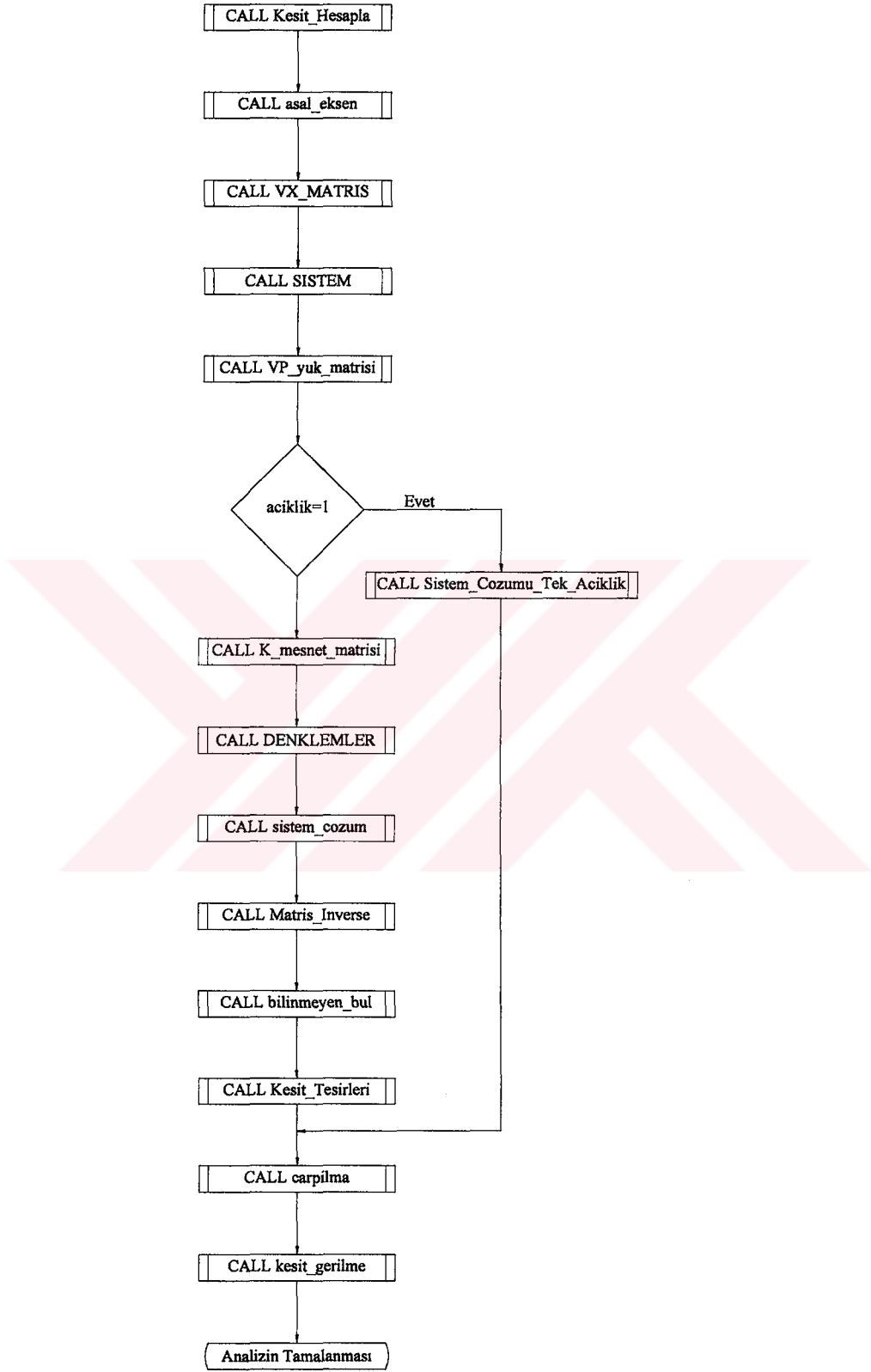
kns=43



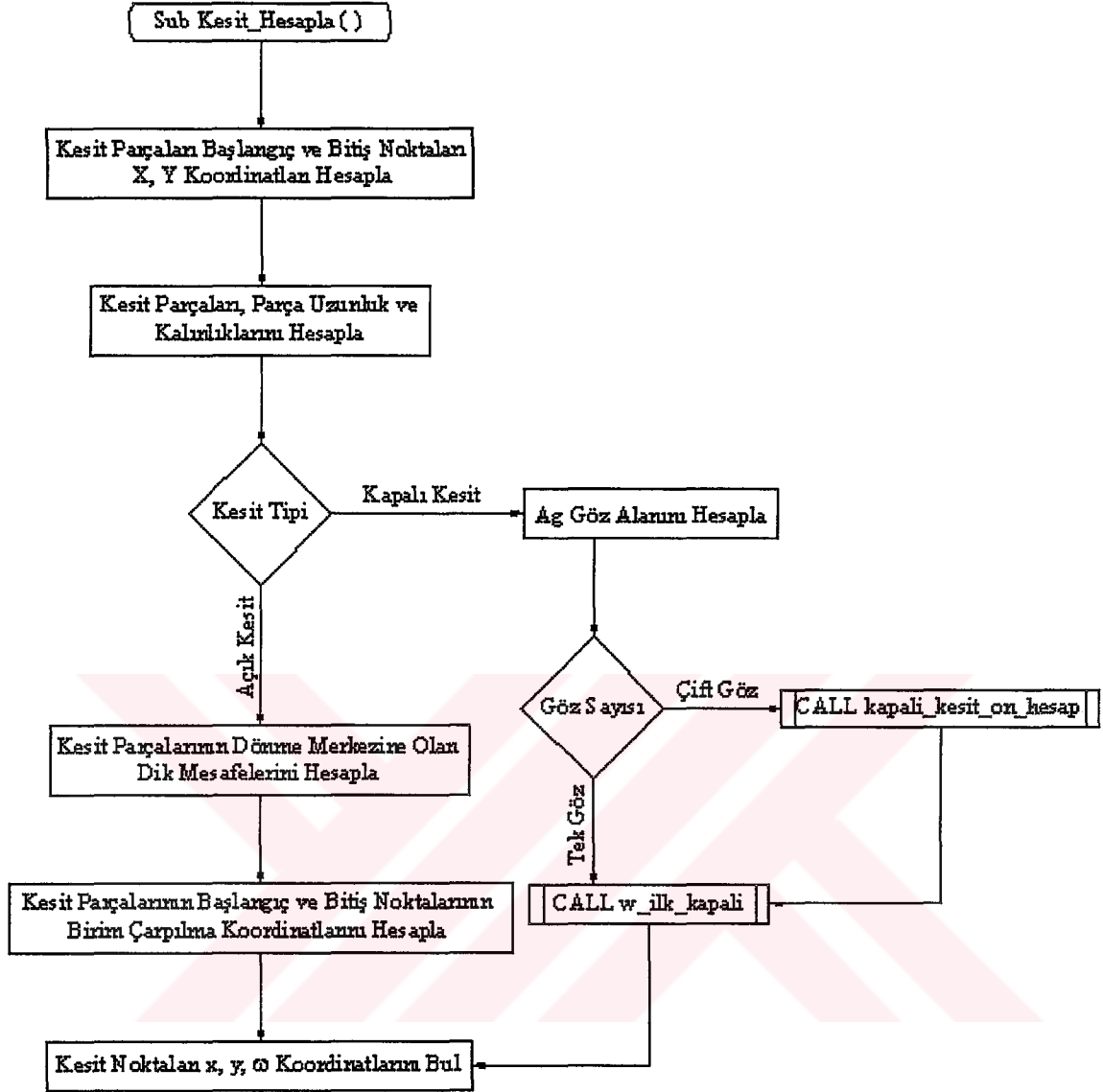
ktn=53

Şekil 6.17 – Çift gözlü kapalı kesit bilgileri

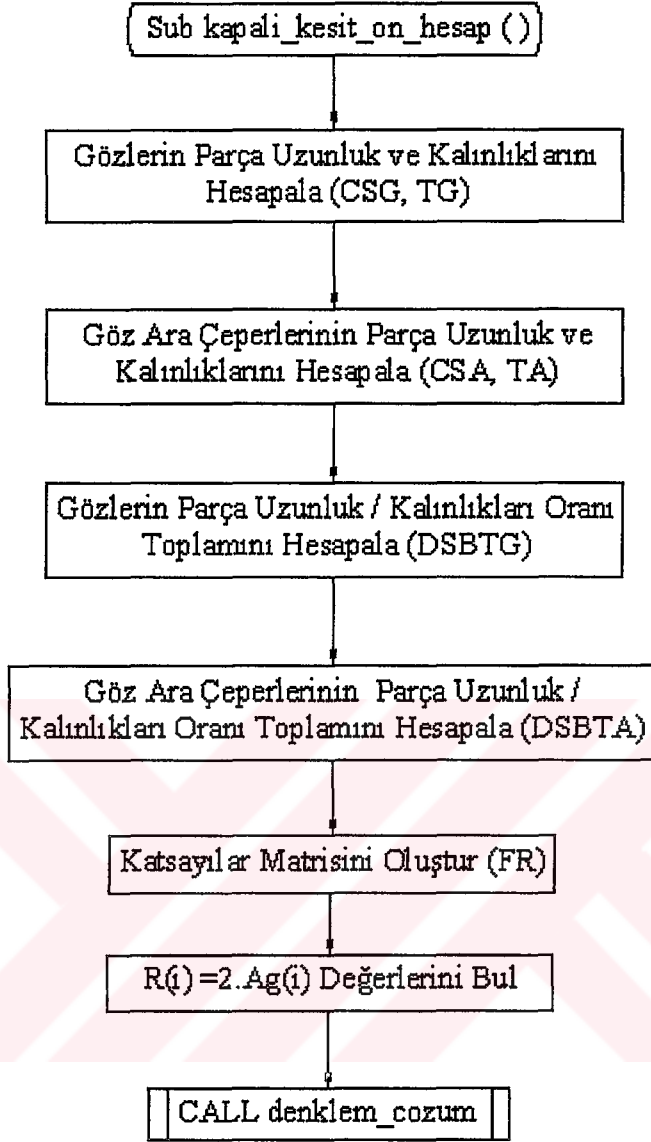
### 6.3.2 TWS Programı Analiz Akış Diyagramları



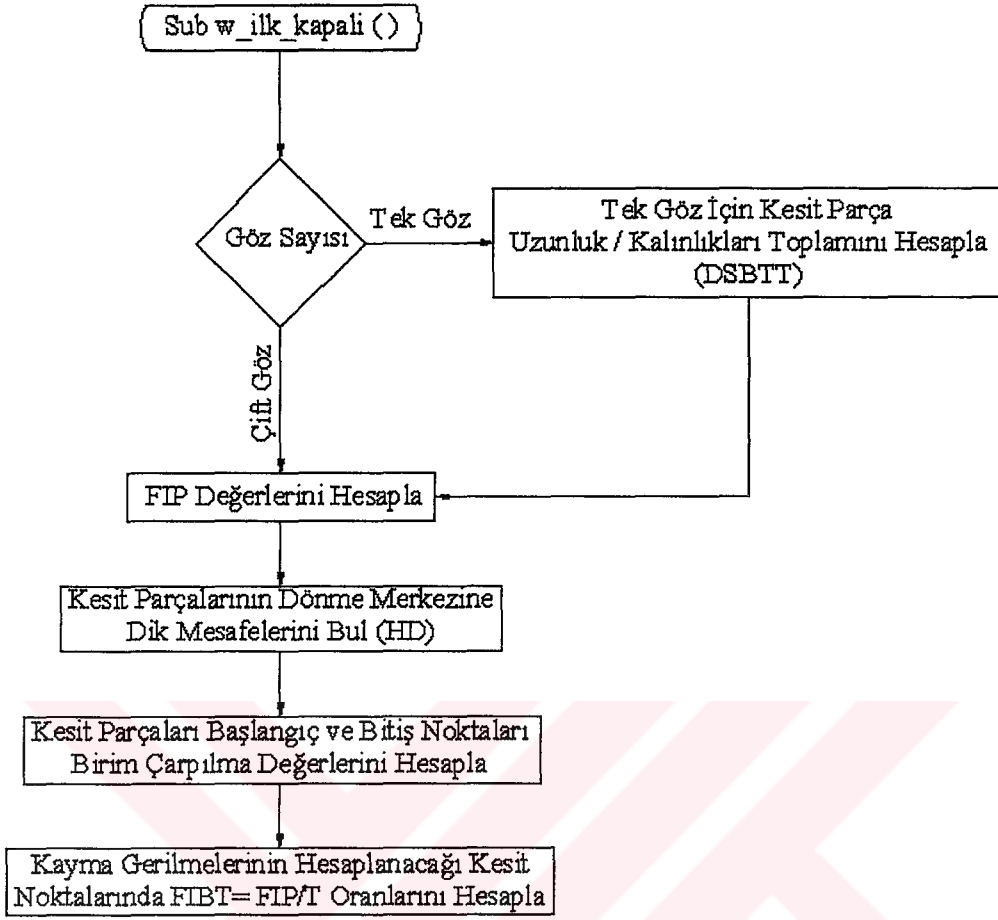
Şekil 4.10 – “Main” Alt Programı Akış Şeması



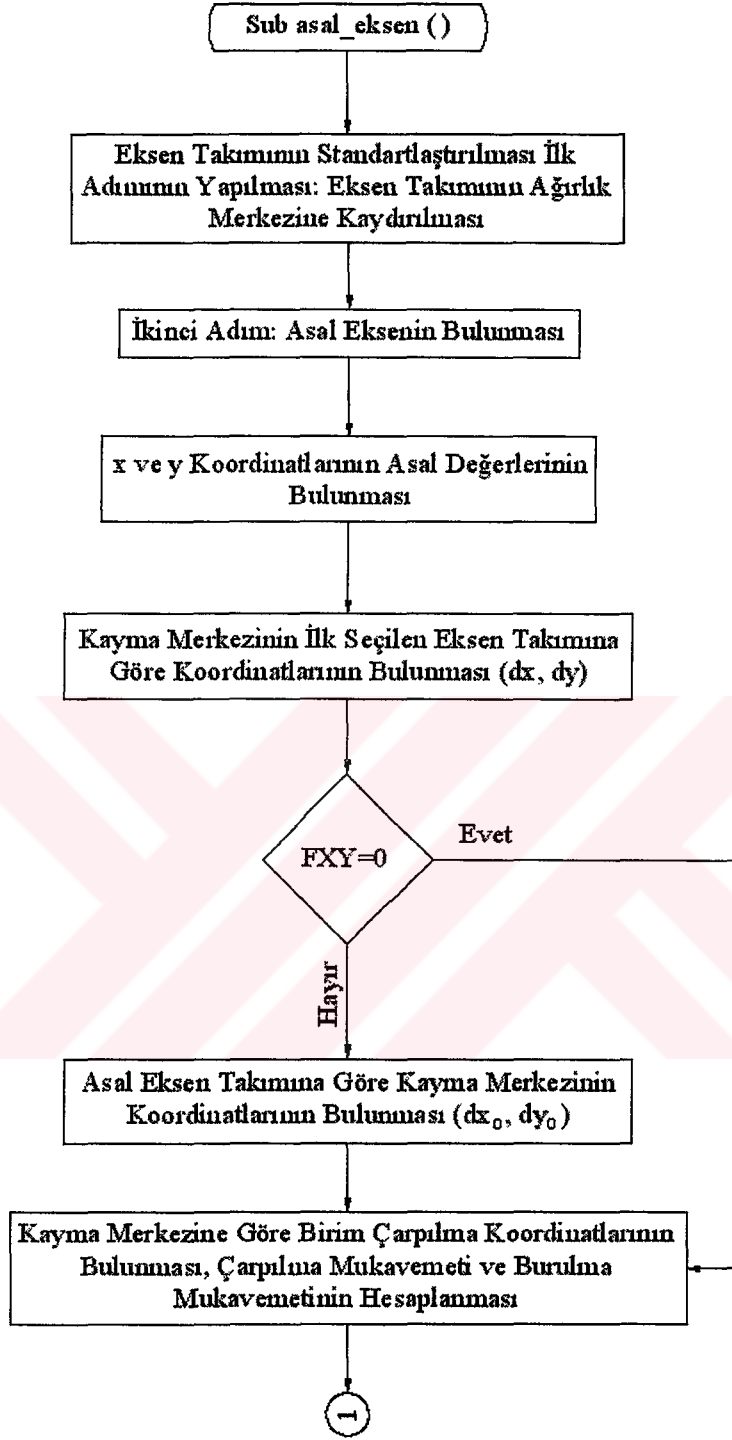
Şekil 4.11 – “Kesit\_Hesapla” Alt Programı Akış Diyagramı

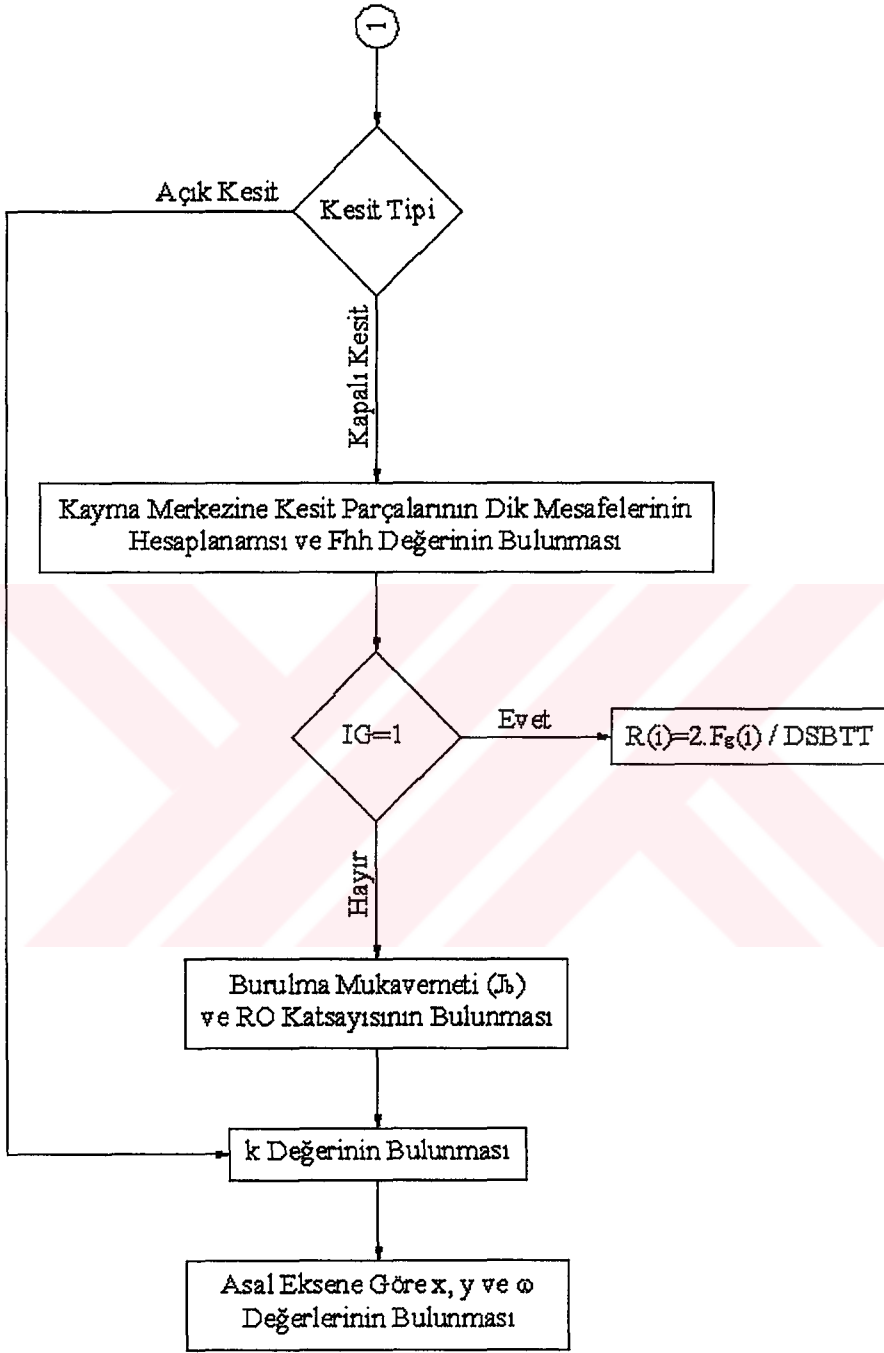


Şekil 4.12 – “kapali\_kesit\_on\_hesap” alt programı akış diyagramı

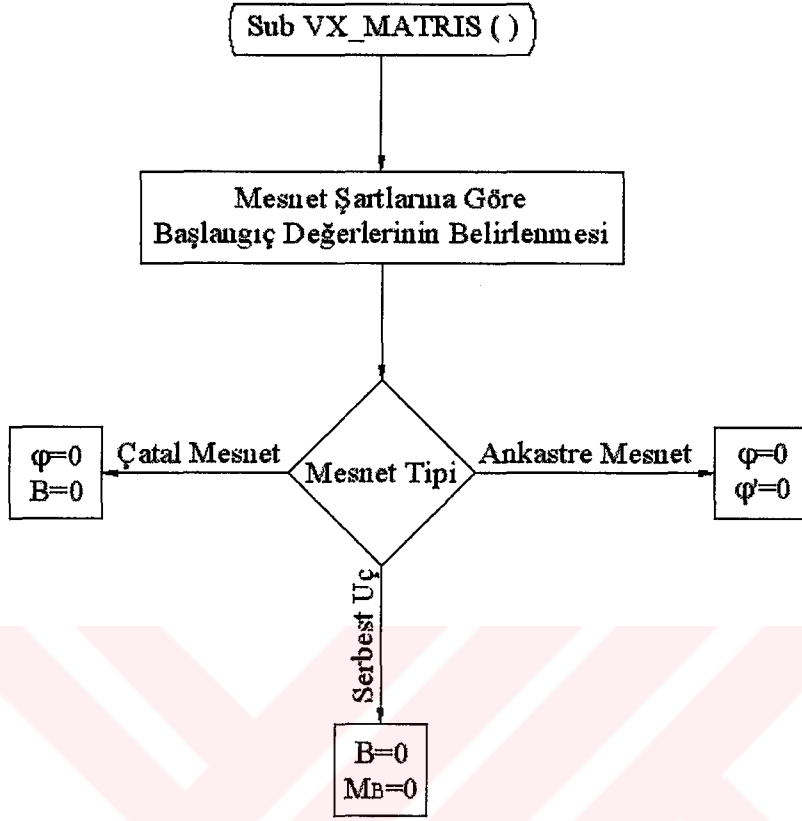


Şekil 4.13 – “w\_ilk\_kapali” alt programı akış diyagramı

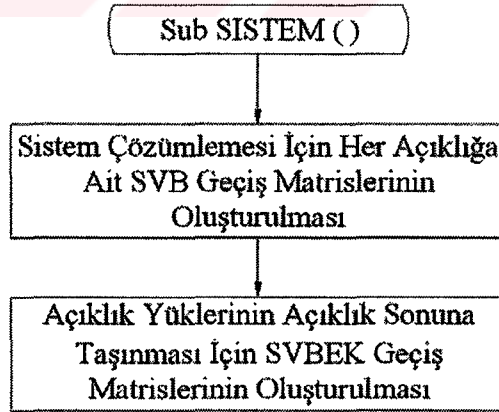




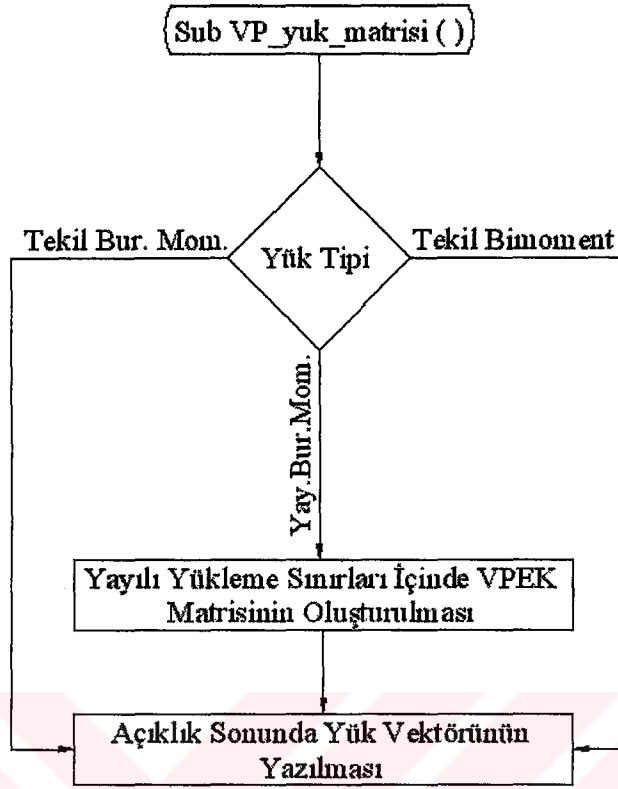
Şekil 4.14 – “asal\_eksen” Alt Programı Akış Diyagramı



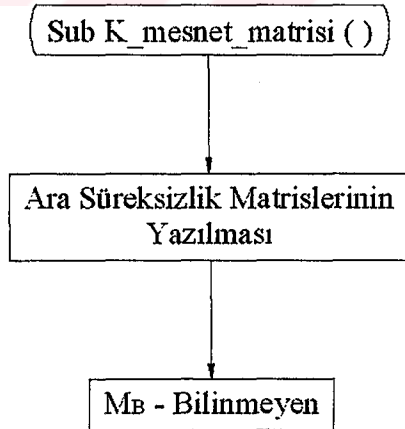
Şekil 4.15 – “VX\_MATRIS” alt programı akış diyagramı



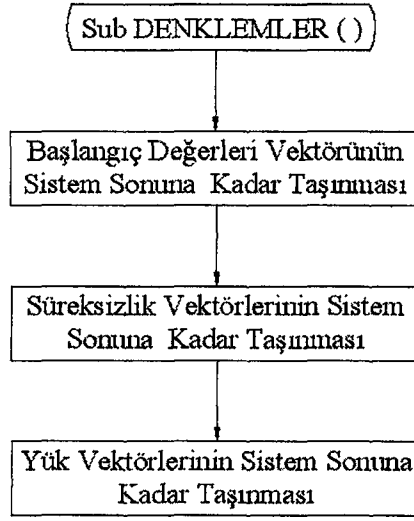
Şekil 4.16 – “SISTEM” alt programı akış diyagramı



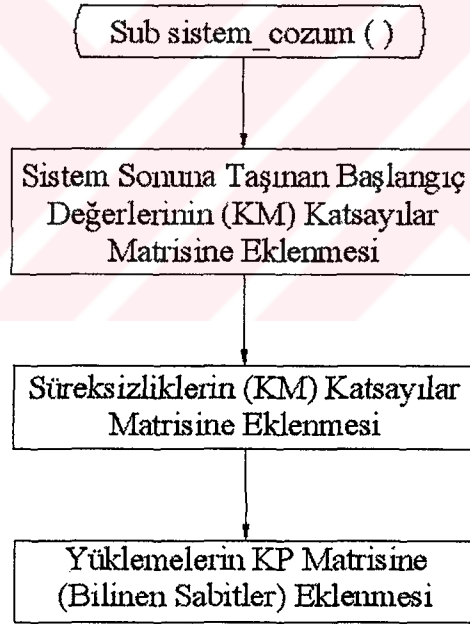
Şekil 4.17 – “VP\_yuk\_matrisi” alt programının akış diyagramı



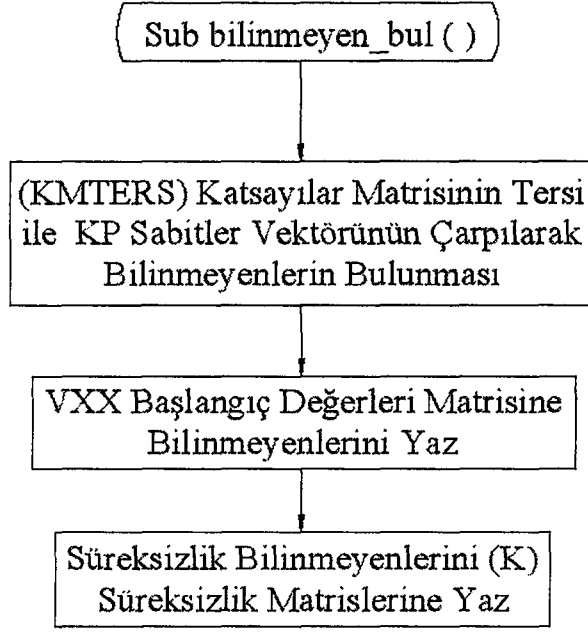
Şekil 4.18 – “K\_sureksizlik\_matrisi” alt programının akış diyagramı



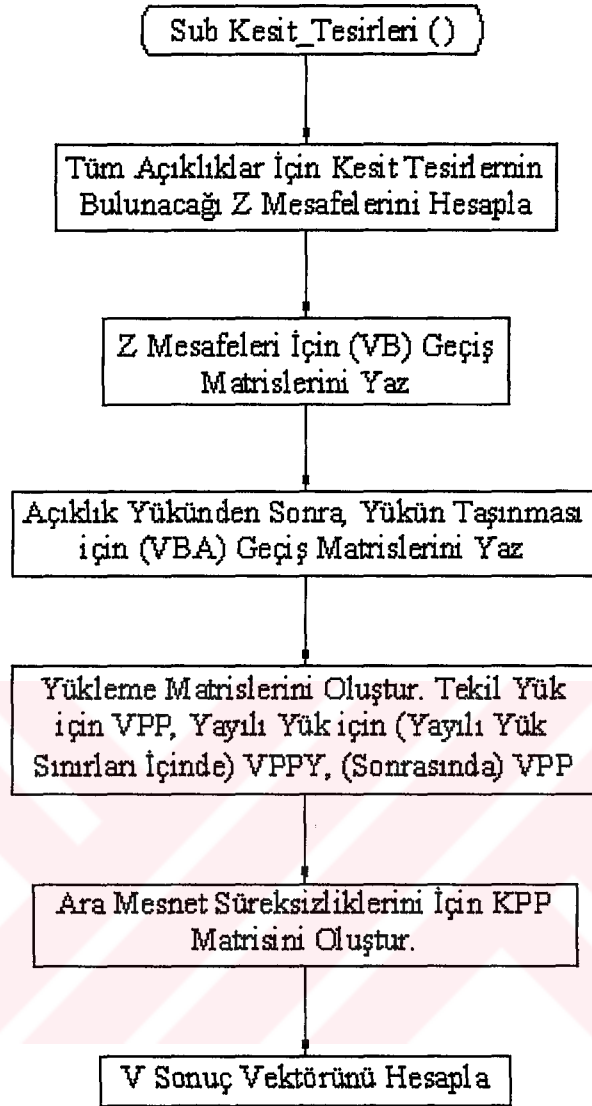
Şekil 4.19 – “DENKLEMLER” alt programının akış diyagramı



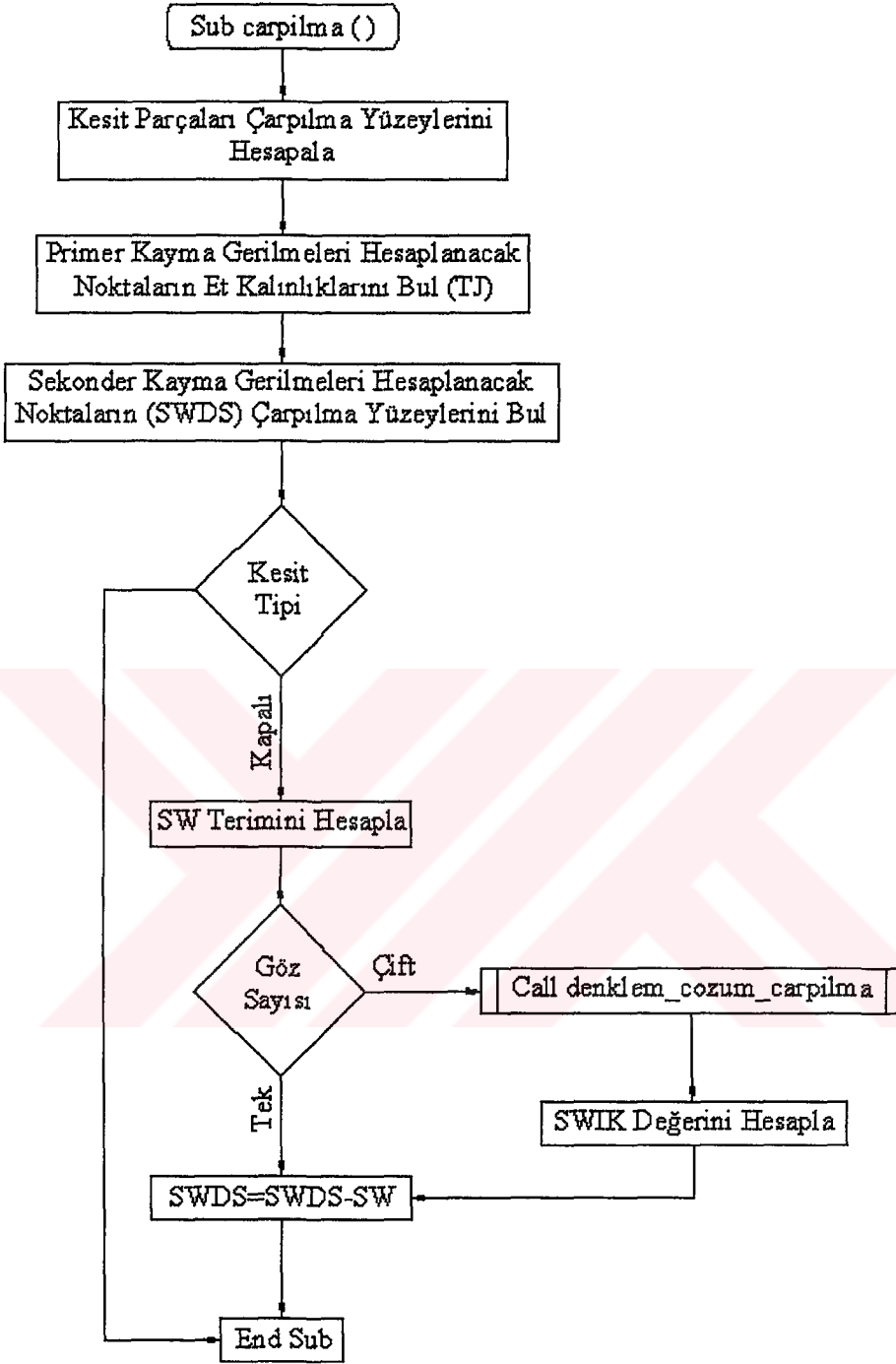
Şekil 4.20 – “sistem\_cozum” alt programının akış diyagramı



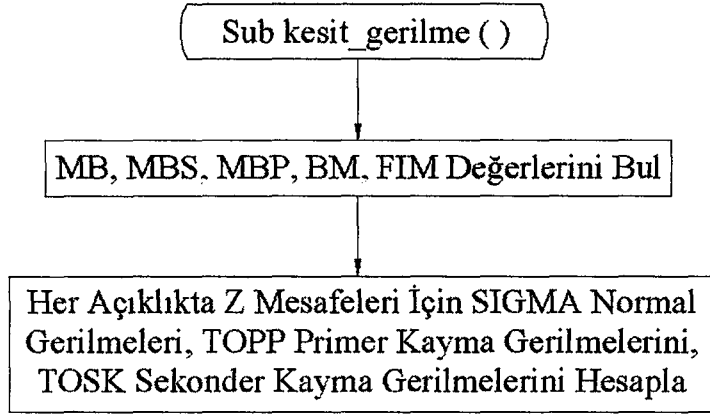
Şekil 4.21 – “bilinmeyen\_bul” alt programının akış diyagramı



Şekil 4.22 – “Kesit\_Tesirleri” alt programının akış diyagramı



Şekil 4.23 – “carpılma” alt programının akış diyagramı



Şekil 4.24 – “kesit\_gerilme” alt programının akış diyagramı



**KAYNAKLAR**

Deren, H. (1983), Çelik Kirişlerin Burulma Yüklerine Göre Hesabı, Çelik Yapılar Seminer Notları Cilt 2, İstanbul Teknik Üniversitesi, İstanbul

Gülek, E. (1985), İnce cidarlı Mütemadi Kirişlerde Burulma Tesirinin İncelenmesi, Yıldız Teknik Üniversitesi, Yüksek Lisans Tezi, İstanbul

İnan, M. (1963), Elastomekanikte Başlangıç Değerleri Metodu ve Taşıma Matrisi, İstanbul Teknik Üniversitesi, İstanbul

İnan, M. (1973), Cisimlerin Mukavemeti, Ofset Matbaacılık, İstanbul

Öztürk, A.Z., (1979), İnce Cidarlı Çubukların Çarpımalı Burulmaya Göre Hesabı ve Stabilitate Problemleri, Yıldız Teknik Üniversitesi, Doçentlik Tezi, İstanbul



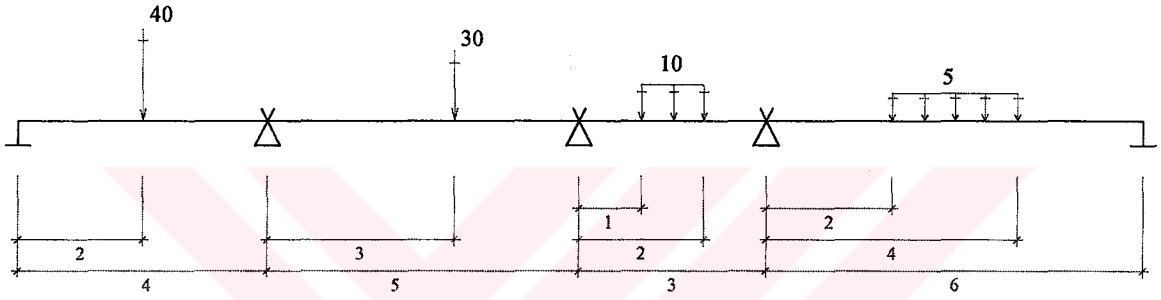
**EKLER**

- Ek 1 Sayısal Örnek - TWS Programı Analiz Sonuçları  
Ek 2 TWS Programı Analiz Kodları

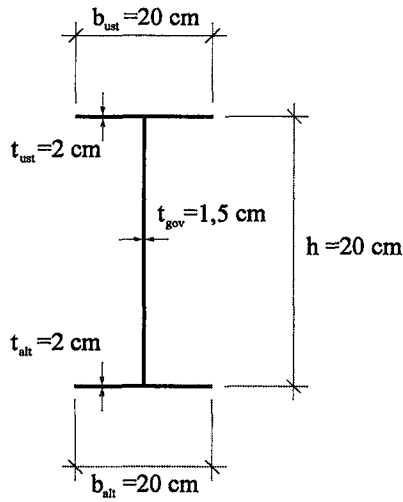


### Ek 1 Sayısal Örnek – TWS Programı Analiz Sonuçları

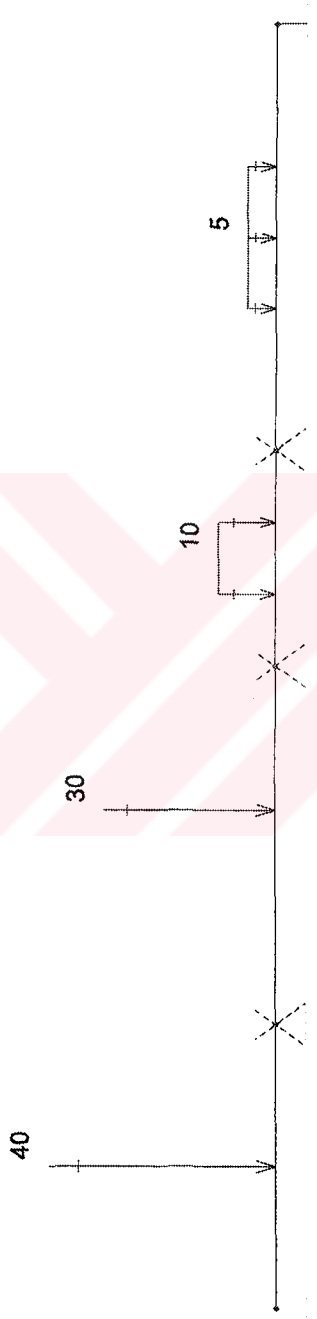
**Örnek 1:** Şekil Ek 1.1’de görülen 4 açıklığa sahip olan sistemde I enkesit kullanılmıştır. Enkesitin başlık genişliği 20 cm, başlık kalınlığı 2 cm, gövde yüksekliği 40 cm, gövde kalınlığı 1,5 cm dir. Mesnet şartları ve yükleme durumları şekil Ek1.1’de görülen sistemin analizi TWS programı yardımı ile yapılmış ve sonuç çıktıları aşağıda verilmiştir. Enkesit gerilme diyagramları 1. açıklıkta  $Z=2$  m için verilmiştir.



Şekil Ek 1.1 – Birinci örnek sistemi



Şekil Ek 1.2 – Enkesit bilgileri



**Sistem Bilgileri:**

Çalışma Birimi : t - m  
Açıklık Sayısı : 4

**Açıklık Mesafeleri:**

1. Açıklık : 4 m  
2. Açıklık : 5 m  
3. Açıklık : 3 m  
4. Açıklık : 6 m

**Mesnet Şartları:**

1. Mesnet : Ankastre Mesnet  
2. Mesnet : Çatal Mesnet  
3. Mesnet : Çatal Mesnet  
4. Mesnet : Çatal Mesnet  
5. Mesnet : Ankastre Mesnet

**Yükleme Bilgileri:**

Açıklık	Yükleme Tipi	Yükleme Değeri		Za		Cz	
1. Açıklık	Tekil Burulma Momenti	40	t-m	2	m	-	
2. Açıklık	Tekil Burulma Momenti	30	t-m	3	m	-	
3. Açıklık	Yayıllı Burulma Momenti	10	t-m	2	m	1	m
4. Açıklık	Yayıllı Burulma Momenti	5	t-m	4	m	2	m

**Enkesit Bilgileri:**

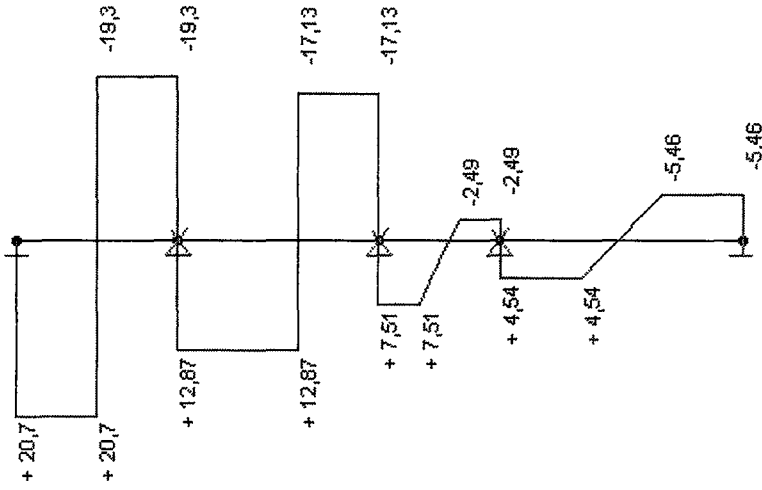
Kesit Tipi : I Kesit  
Üst Başlık Genişliği (bust) : 0,2 m  
Üst Başlık Kalınlığı (tust) : 0,02 m  
Gövde Yüksekliği (h) : 0,4 m  
Gövde Kalınlığı (tgov) : 0,015 m  
Alt Başlık Genişliği (balt) : 0,2 m  
Alt Başlık Kalınlığı (talt) : 0,02 m

**Enkesit Değerleri:**

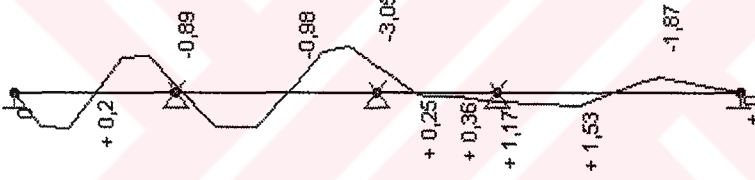
Enkesit Alanı : A : 0,014 m<sup>2</sup>  
X Eksenine Göre Atalet Momenti : F<sub>xx</sub> : 2,66666666666667E-05 m<sup>4</sup>  
Y Eksenine Göre Atalet Momenti : F<sub>yy</sub> : 0,0004 m<sup>4</sup>  
Çarpılma Mukavemeti : F<sub>ωω</sub> : 1,06666666666667E-06 m<sup>6</sup>  
Burulma Atalet Momenti : J<sub>b</sub> : 1,51666666666667E-06 m<sup>4</sup>  
k : 0,735980072193987

Z (m)	$\varphi$	M <sub>BP</sub> (tm)	M <sub>BS</sub> (tm)	M <sub>B</sub> (tm)	B (tm <sup>2</sup> )
1. Açıklık					
0	0	0	20,699	20,699	-17,758
0,667	0,134	4,131	16,568	20,699	-5,579
1,333	0,38	4,193	16,506	20,699	5,23
2	0,521	0,201	20,499	20,699	17,323
2	0,521	0,201	-19,501	-19,301	17,323
2,667	0,4	-3,914	-15,386	-19,301	5,922
3,333	0,159	-4,25	-15,051	-19,301	-4,025
4	0	-0,889	-18,411	-19,301	-14,96
2. Açıklık					
0	0	-0,889	13,756	12,867	-14,96
1	0,171	4,069	8,797	12,867	-4,167
2	0,536	4,043	8,823	12,867	4,267
3	0,702	-0,982	13,848	12,867	15,117
3	0,702	-0,982	-16,152	-17,133	15,117
3,667	0,532	-4,679	-12,455	-17,133	5,768
4,333	0,244	-5,317	-11,816	-17,133	-2,164
5	0	-3,053	-14,081	-17,133	-10,627
3. Açıklık					
0	0	-3,053	10,558	7,505	-10,627
1	-0,084	0,247	7,259	7,505	-2,1
1,25	-0,077	0,423	4,582	5,005	-0,624
1,5	-0,068	0,445	2,06	2,505	0,204
1,75	-0,059	0,396	-0,391	0,005	0,412
2	-0,051	0,361	-2,856	-2,495	0,007
3	-1,236	1,165	-3,66	-2,495	-3,111
4. Açıklık					
0	-1,236	1,165	3,373	4,538	-3,111
2	-0,936	1,529	3,009	4,538	2,323
2,5	-0,888	0,736	1,302	2,038	3,389
3	-0,878	-0,235	-0,227	-0,462	3,655
3,5	-0,907	-1,175	-1,787	-2,962	3,157
4	-0,971	-1,87	-3,592	-5,462	1,827
6	-3,709	0	-5,462	-5,462	-5,882

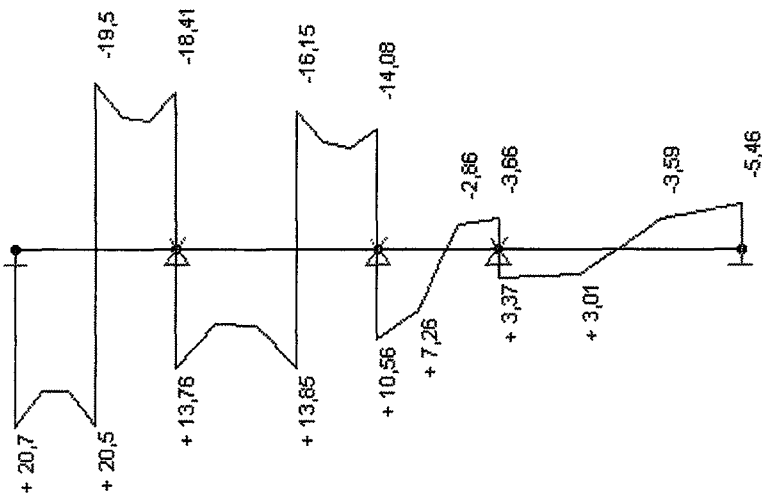
NOT: Tabloda Tekrar Eden ( Z ) Değerleri Tekil Yükleminin Sol ve Sağındaki Noktalardır.



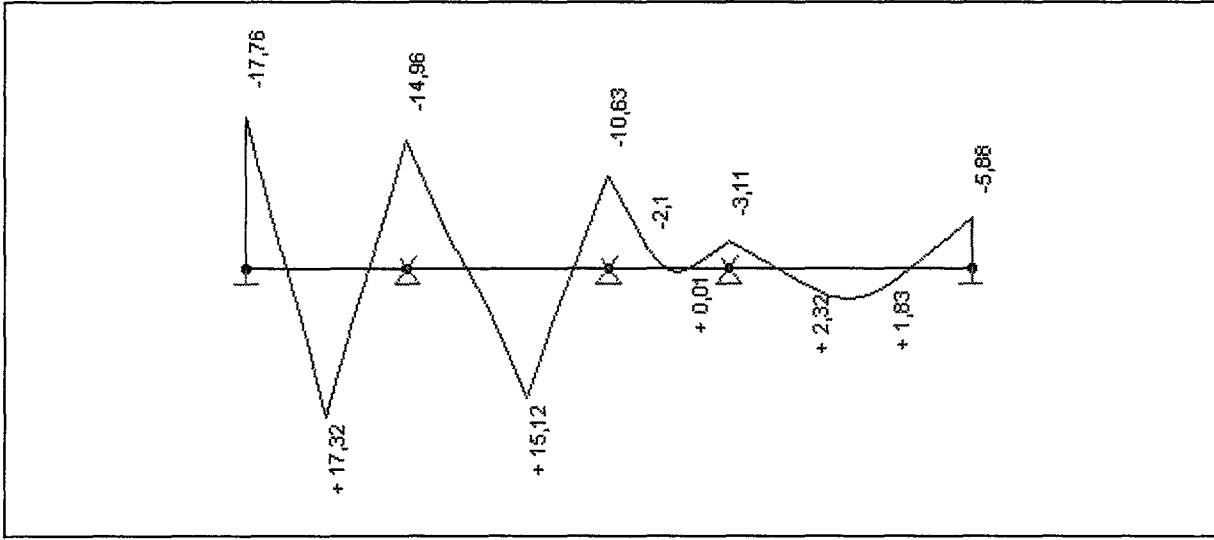
Diyagram 1 - Toplam Burulma Momenti Diyagramı



Diyagram 2 - Primer Burulma Momenti Diyagramı

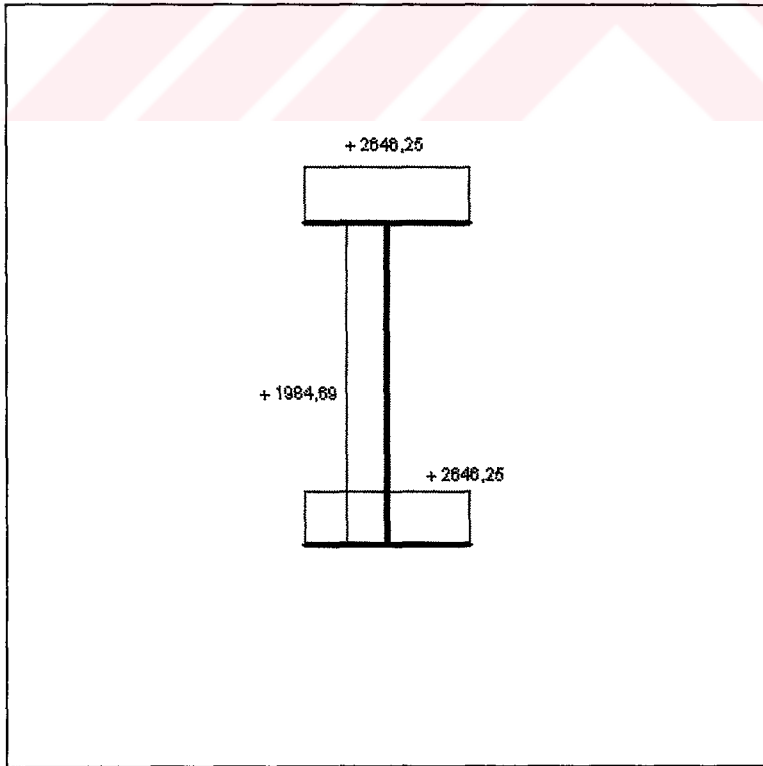
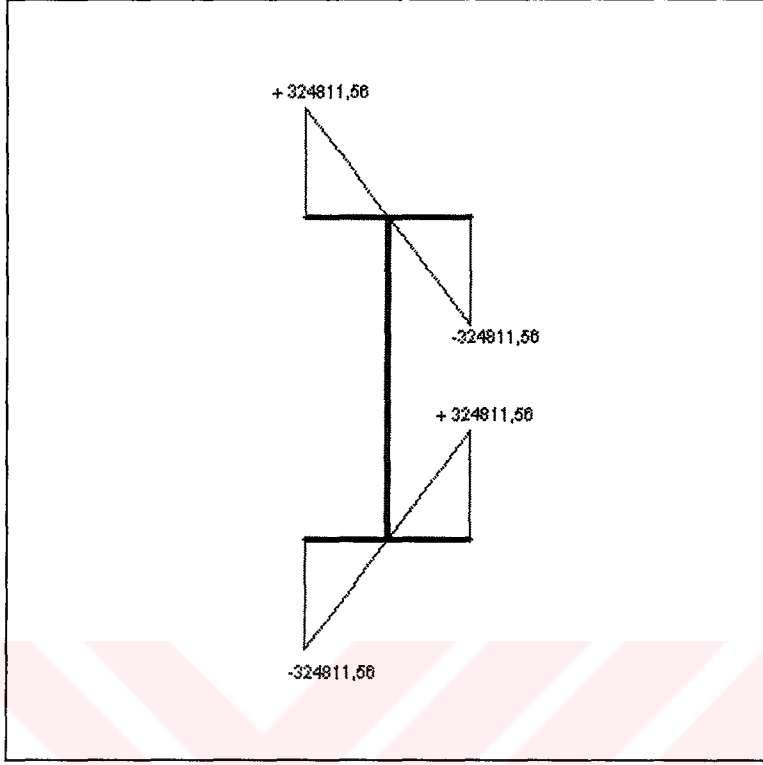


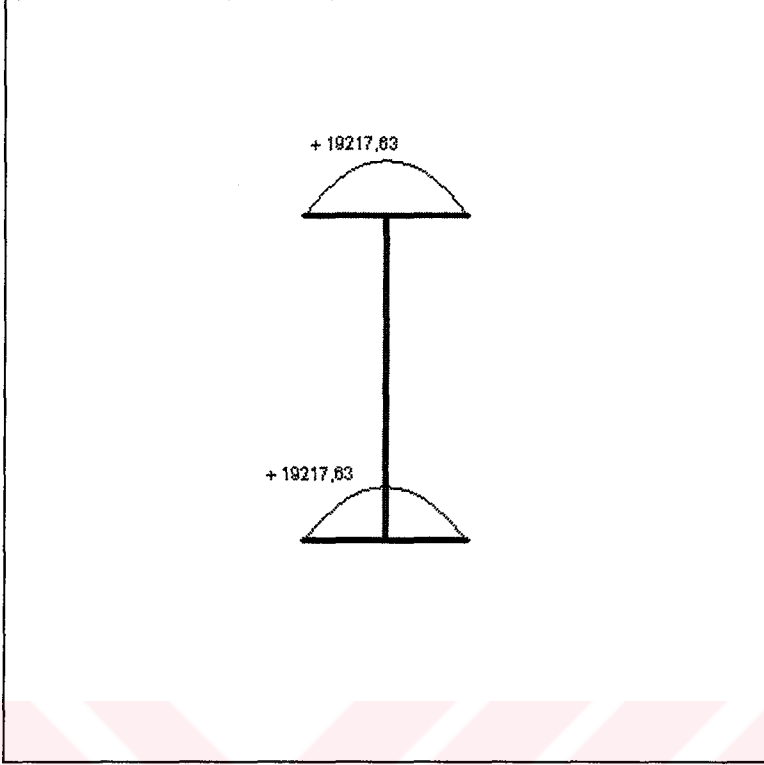
Diyagram 3 - Sekonder Burulma Momenti Diyagramı



Diyagram 4 - Bimoment Diyagramı

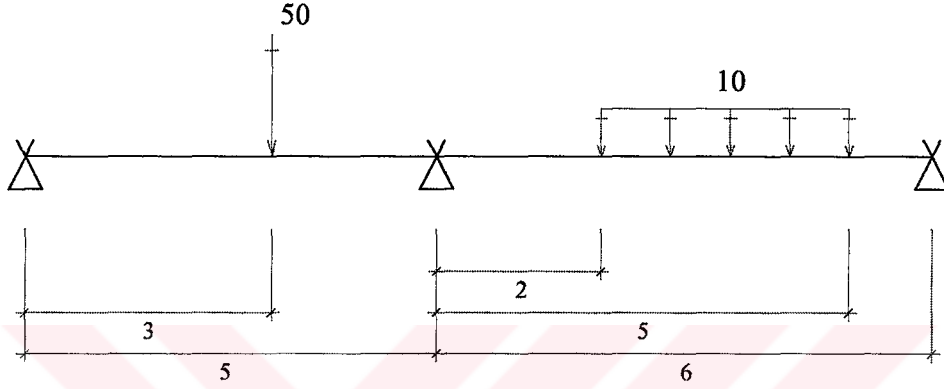


**Enkesit Gerilme Diyagramları: 1. Açıklık, Z = 2**

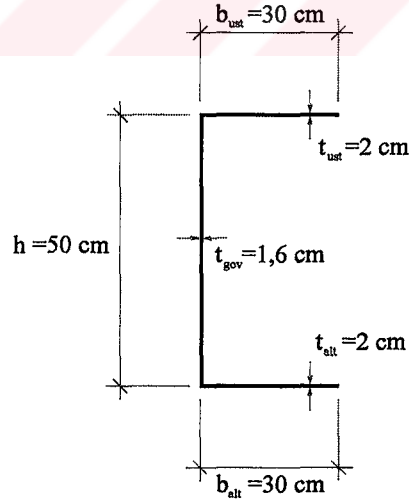


Sekonder Kayma Gerilmesi Diyagramı

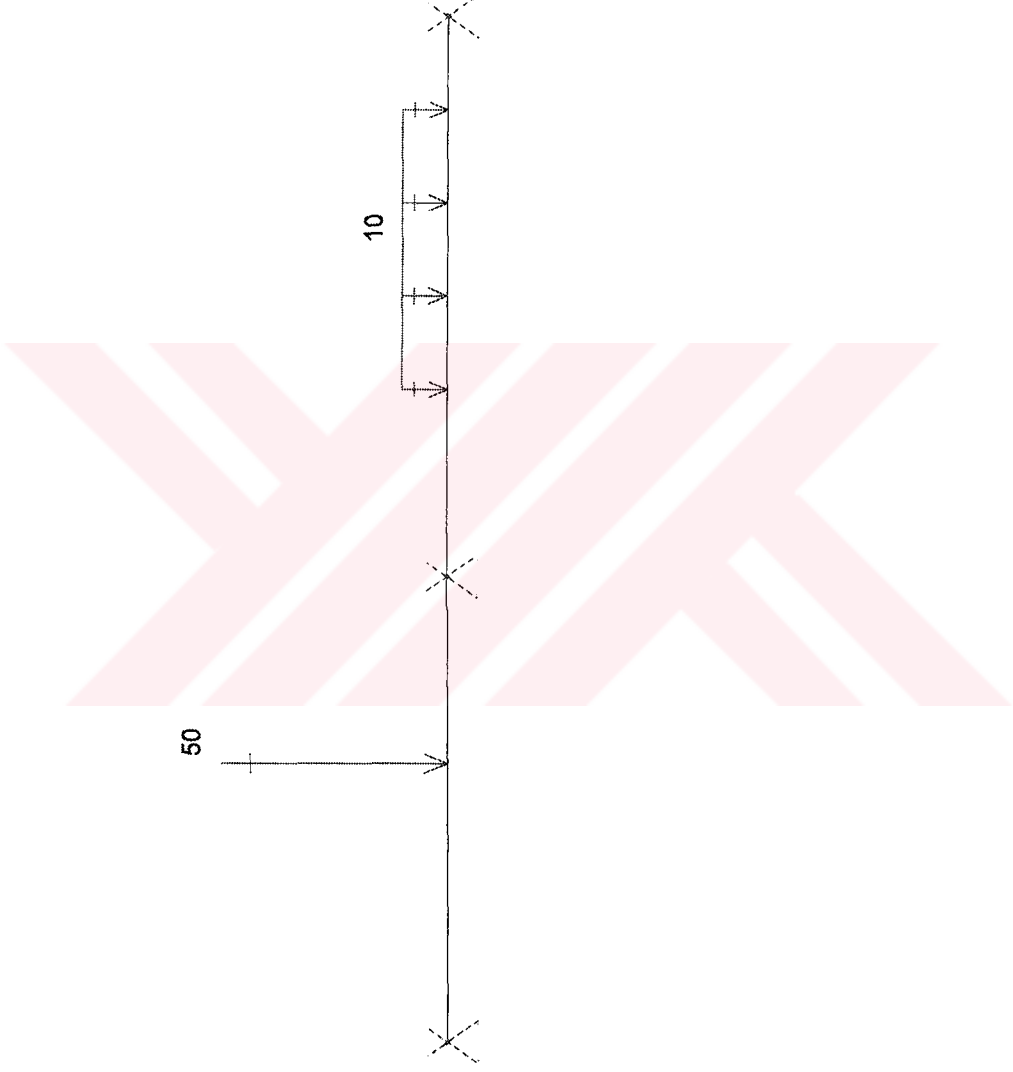
**Örnek 2:** Şekil Ek 1.2 'de görülen 2 açıklığa sahip olan sistemde C enkesit kullanılmıştır. Enkesitin başlık genişliği 30 cm, başlık kalınlığı 2 cm, gövde yüksekliği 50 cm, gövde kalınlığı 1,6 cm dir. Mesnet şartları ve yükleme durumları şekil Ek1.2'de görülen sistemin analizi TWS programı yardımı ile yapılmış ve sonuç çıktıları aşağıda verilmiştir. Enkesit gerilme diyagramları 1. açıklıkta  $Z=3$  m için verilmiştir.



Şekil Ek 1.3 – İkinci örnek sistemi



Şekil Ek 1.4 – Enkesit Bilgileri



**Sistem Bilgileri:**

Çalışma Birimi : t - m  
Açıklık Sayısı : 2

**Açıklık Mesafeleri:**

1. Açıklık : 5 m  
2. Açıklık : 6 m

**Mesnet Şartları:**

1. Mesnet : Çatal Mesnet  
2. Mesnet : Çatal Mesnet  
3. Mesnet : Çatal Mesnet

**Yükleme Bilgileri:**

Açıklık	Yükleme Tipi	Yükleme Değeri		Za		Cz	
1. Açıklık	Tekil Burulma Momenti	50	t-m	3	m	-	
2. Açıklık	Yayıllı Burulma Momenti	10	t-m	5	m	2	m

**Enkesit Bilgileri:**

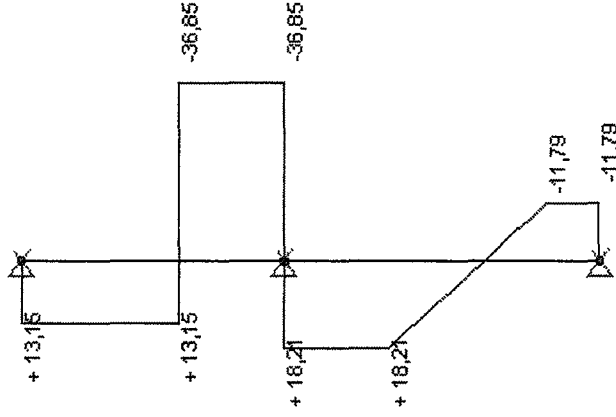
Kesit Tipi : C Kesit  
Üst Başlık Genişliği (bust) : 0,3 m  
Üst Başlık Kalınlığı (tust) : 0,02 m  
Gövde Yüksekliği (h) : 0,5 m  
Gövde Kalınlığı (tgov) : 0,016 m  
Alt Başlık Genişliği (balt) : 0,3 m  
Alt Başlık Kalınlığı (talt) : 0,02 m

**Enkesit Değerleri:**

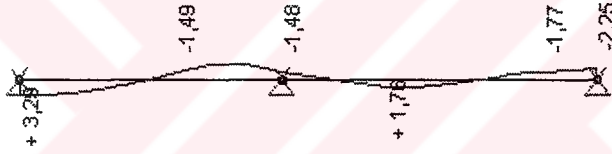
Enkesit Alanı A : 0,02 m<sup>2</sup>  
X Eksenine Göre Atalet Momenti F<sub>xx</sub> : 0,000198 m<sup>4</sup>  
Y Eksenine Göre Atalet Momenti F<sub>yy</sub> : 9,16666666666667E-04 m<sup>4</sup>  
Çarpılma Mukavemeti F<sub>ωω</sub> : 8,69318181818182E-06 m<sup>6</sup>  
Burulma Atalet Momenti J<sub>b</sub> : 2,28266666666667E-06 m<sup>4</sup>  
k : 0,316276710442316

Z (m)	$\varphi$	M <sub>BP</sub> (tm)	M <sub>BS</sub> (tm)	M <sub>B</sub> (tm)	B (tm <sup>2</sup> )
1. Açıklık					
0	0	3,294	9,858	13,152	0
0,5	0,089	3,17	9,982	13,152	4,95
1	0,171	2,796	10,365	13,152	10,023
1,5	0,24	2,163	10,989	13,152	15,348
2	0,287	1,255	11,897	13,152	21,058
2,5	0,306	0,048	13,104	13,152	27,295
3	0,287	-1,487	14,639	13,152	34,216
3	0,287	-1,487	-35,361	-36,848	34,216
3,333	0,251	-2,433	-34,415	-36,848	22,598
3,667	0,2	-2,997	-33,851	-36,848	11,23
4	0,143	-3,184	-33,665	-36,848	-0,012
4,333	0,086	-2,996	-33,852	-36,848	-11,254
4,667	0,036	-2,432	-34,416	-36,848	-22,622
5	0	-1,485	-35,364	-36,848	-34,241
2. Açıklık					
0	0	-1,485	19,691	18,207	-34,241
2	0,078	1,759	16,448	18,207	0,739
2,3	0,107	1,667	13,54	15,207	5,234
2,6	0,133	1,454	10,753	12,207	8,875
2,9	0,154	1,143	8,064	9,207	11,696
3,2	0,17	0,76	5,447	6,207	13,721
3,5	0,179	0,327	2,88	3,207	14,969
3,8	0,18	-0,131	0,338	0,207	15,451
4,1	0,174	-0,593	-2,201	-2,793	15,172
4,4	0,161	-1,034	-4,759	-5,793	14,129
4,7	0,141	-1,433	-7,36	-8,793	12,313
5	0,114	-1,765	-10,028	-11,793	9,706
6	-1,369	-2,247	-9,546	-11,793	0

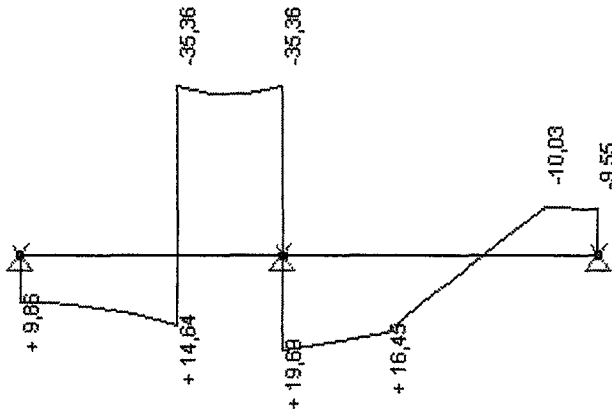
NOT: Tabloda Tekrar Eden ( Z ) Değerleri Tekil Yükleminin Sol ve Sağındaki Noktalardır.



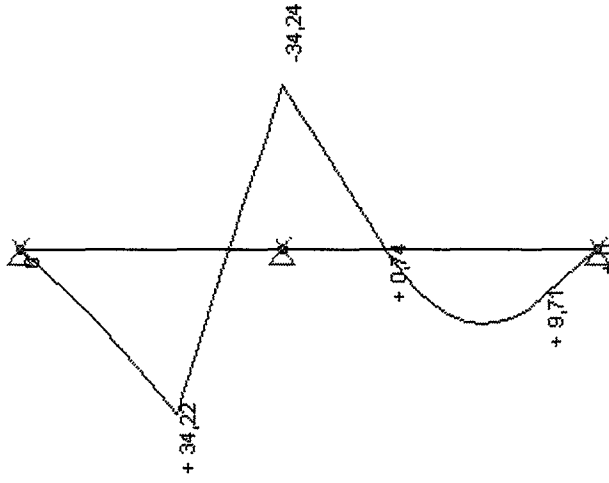
Diyagram 1 - Toplam Burulma Momenti Diyagramı



Diyagram 2 - Primer Burulma Momenti Diyagramı

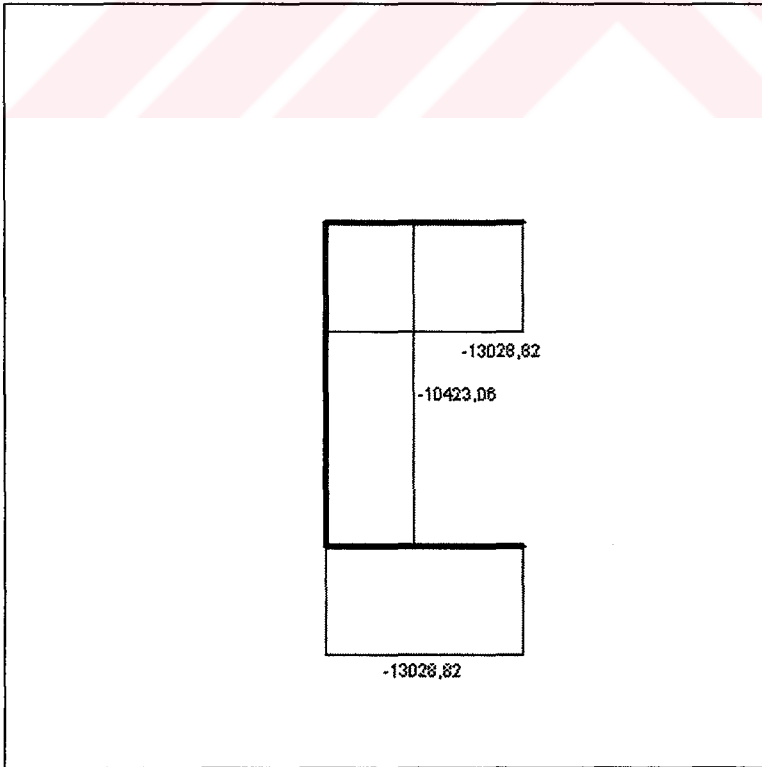
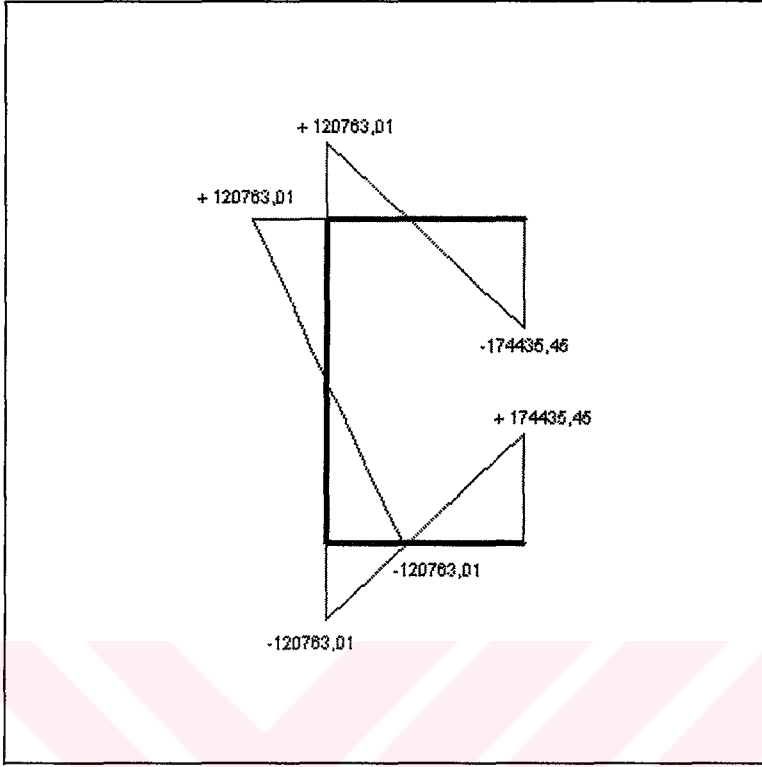


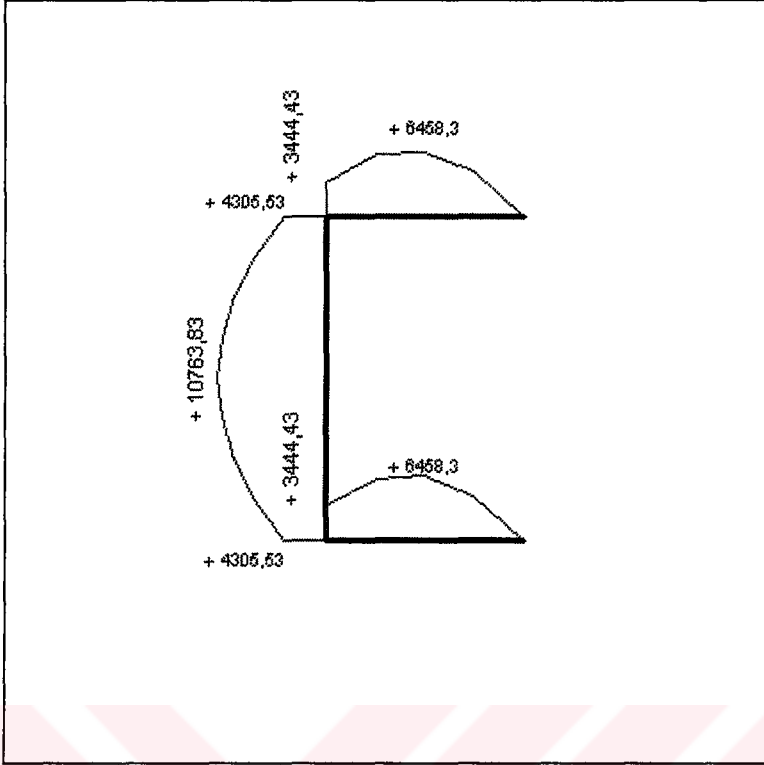
Diyagram 3 - Sekonder Burulma Momenti Diyagramı



Diyagram 4 - Bimoment Diyagramı

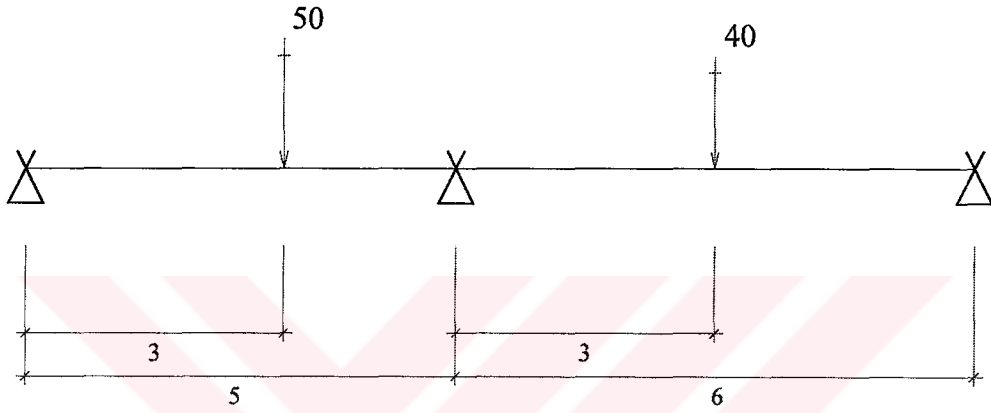


**Enkesit Gerilme Diyagramları: 1. Açıklık, Z = 3**

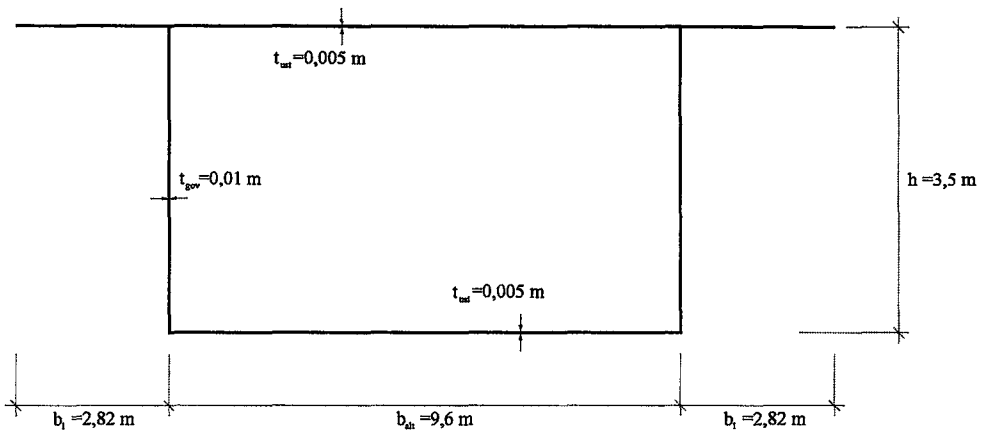


Sekonder Kayma Gerilmesi Diyagramı

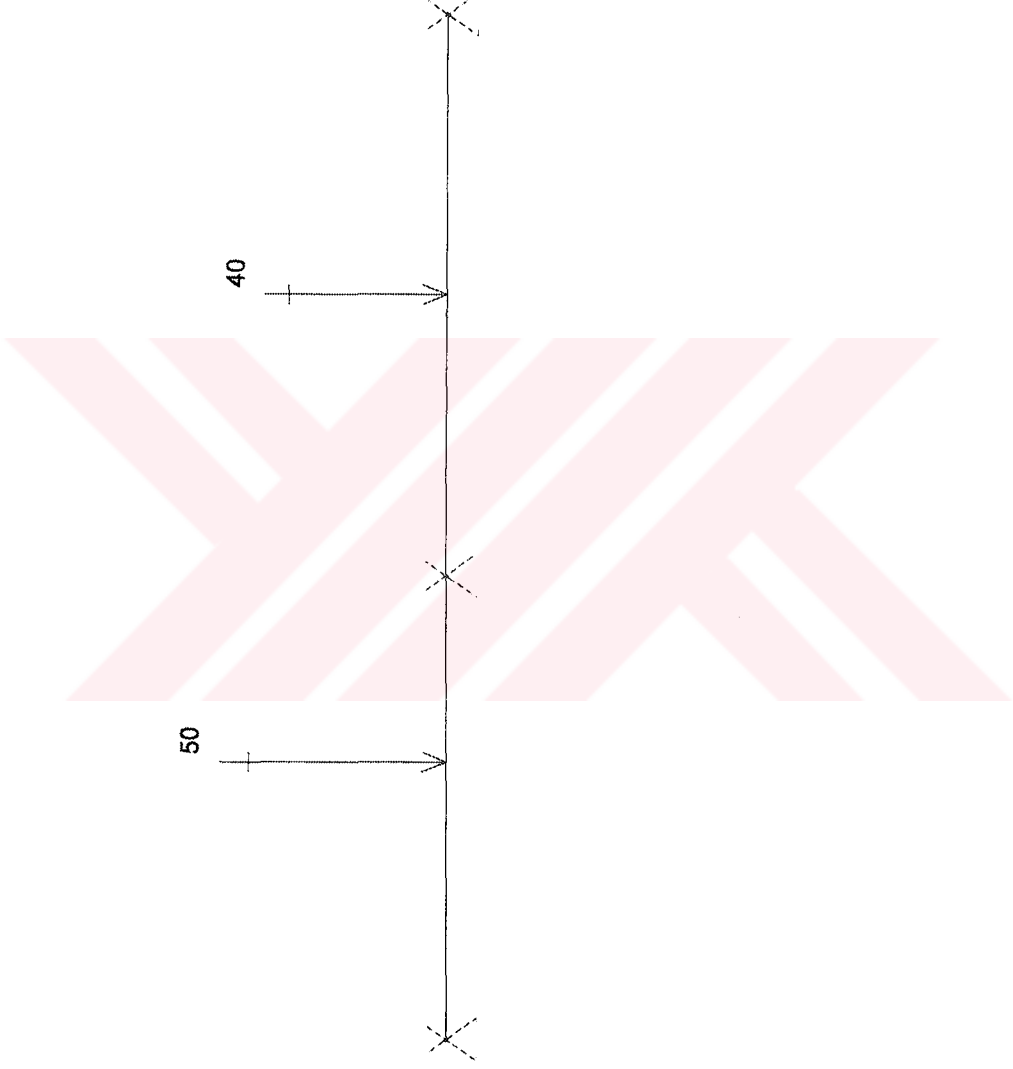
**Örnek 3:** Şekil Ek 1.3 'de görülen 2 açıklığa sahip olan sistemde tek gözlü kapalı kesit kullanılmıştır. Enkesitin alt başlık genişliği 9,6 m, alt ve üst başlık kalınlığı 0,005 m, gövde yüksekliği 3,5 m, gövde kalınlığı 0,01 m, üst başlık açık kısım genişliği 2,82 m dir. Mesnet şartları ve yükleme durumları şekil Ek1.3'de görülen sistemin analizi TWS programı yardımı ile yapılmış ve sonuç çıktıları aşağıda verilmiştir. Enkesit gerilme diyagramları 1. açıklıkda  $Z=2$  m için verilmiştir.



Şekil Ek 1.5 – Üçüncü örnek sistemi



Şekil Ek 1.6 – Kesit bilgileri



**Sistem Bilgileri:**

Çalışma Birimi : t - m  
Açıklık Sayısı : 2

**Açıklık Mesafeleri:**

1. Açıklık : 5 m  
2. Açıklık : 6 m

**Mesnet Şartları:**

1. Mesnet : Çatal Mesnet  
2. Mesnet : Çatal Mesnet  
3. Mesnet : Çatal Mesnet

**Yükleme Bilgileri:**

Açıklık	Yükleme Tipi	Yükleme Değeri		Za		Cz
1. Açıklık	Tekil Burulma Momenti	50	t-m	3	m	-
2. Açıklık	Tekil Burulma Momenti	40	t-m	3	m	-

**Enkesit Bilgileri:**

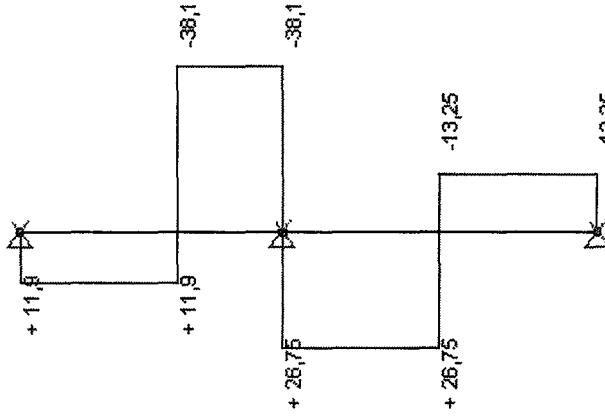
Kesit Tipi :  
Üst Başlık Genişliği (bust) : 15,24 m  
Üst Başlık Kalınlığı (tust) : 0,005 m  
Gövde Yüksekliği (h) : 3,5 m  
Gövde Kalınlığı (tgv) : 0,01 m  
Alt Başlık Genişliği (balt) : 9,6 m  
Alt Başlık Kalınlığı (talt) : 0,005 m

**Enkesit Değerleri:**

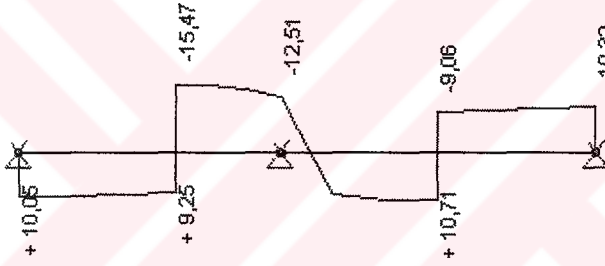
Enkesit Alanı A : 0,2292 m<sup>2</sup>  
X Eksenine Göre Atalet Momenti F<sub>xx</sub> : 3,45627576 m<sup>4</sup>  
Y Eksenine Göre Atalet Momenti F<sub>yy</sub> : 0,476924247382199 m<sup>4</sup>  
Çarpılma Mukavemeti F<sub>ωω</sub> : 2,15152535665185 m<sup>6</sup>  
Burulma Atalet Momenti J<sub>b</sub> : 0,994682949096916 m<sup>4</sup>  
k : 0,2984221126512  
RO : 1,97763844260821  
F<sub>hh</sub> : 2,012117315163

Z (m)	$\varphi$	M <sub>BP</sub> (tm)	M <sub>BS</sub> (tm)	M <sub>B</sub> (tm)	B (tm <sup>2</sup> )
1. Açıklık					
0	0	10,046	1,858	11,904	0
1	0	9,963	1,941	11,904	1,886
2	0	9,705	2,199	11,904	3,941
3	0	9,25	2,654	11,904	6,349
3	0	-15,467	-22,629	-38,096	6,349
3,667	0	-15,397	-22,699	-38,096	-8,71
4,333	0	-14,426	-23,67	-38,096	-24,115
5	0	-12,515	-25,581	-38,096	-40,479
2. Açıklık					
0	0	-12,515	39,261	26,746	-40,479
1	0	9,662	17,084	26,746	-22,146
2	0	10,897	15,849	26,746	-5,8
3	0	10,71	16,036	26,746	10,025
3	0	-9,063	-4,19	-13,254	10,025
4	0	-9,782	-3,472	-13,254	6,222
5	0	-10,188	-3,065	-13,254	2,978
6	0	-10,32	-2,934	-13,254	0

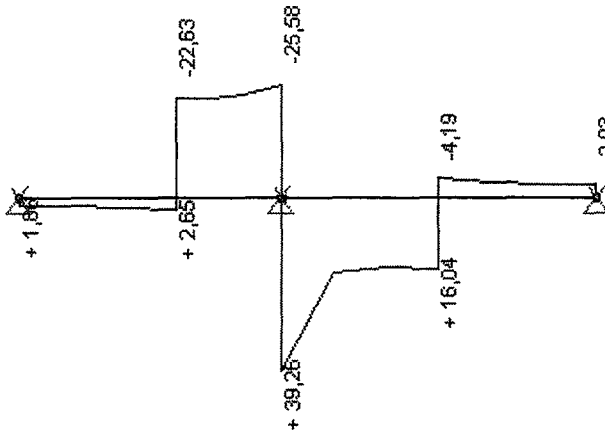
NOT: Tabloda Tekrar Eden ( Z ) Değerleri Tekil Yüklemenin Sol ve Sağındaki Noktalardır.



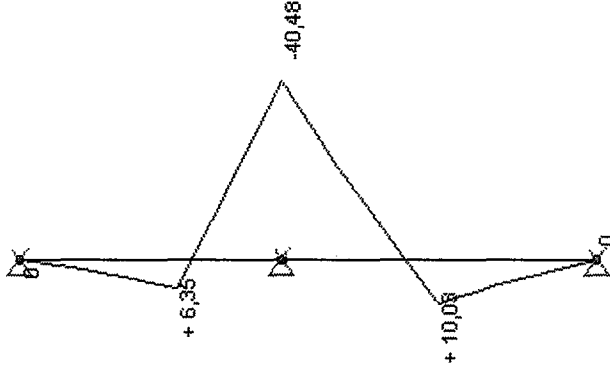
Diyagram 1 - Toplam Burulma Momenti Diyagramı



Diyagram 2 - Primer Burulma Momenti Diyagramı

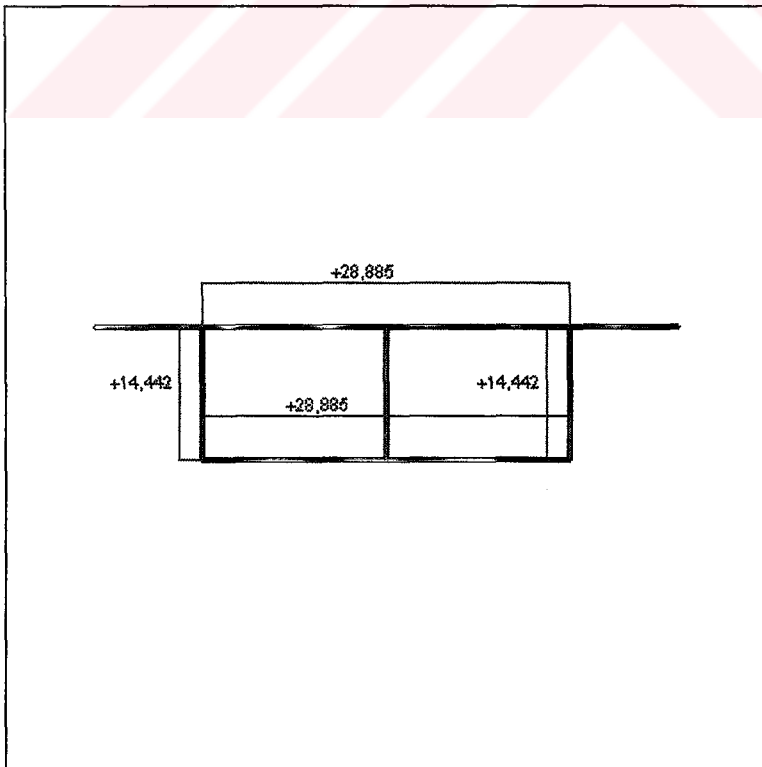
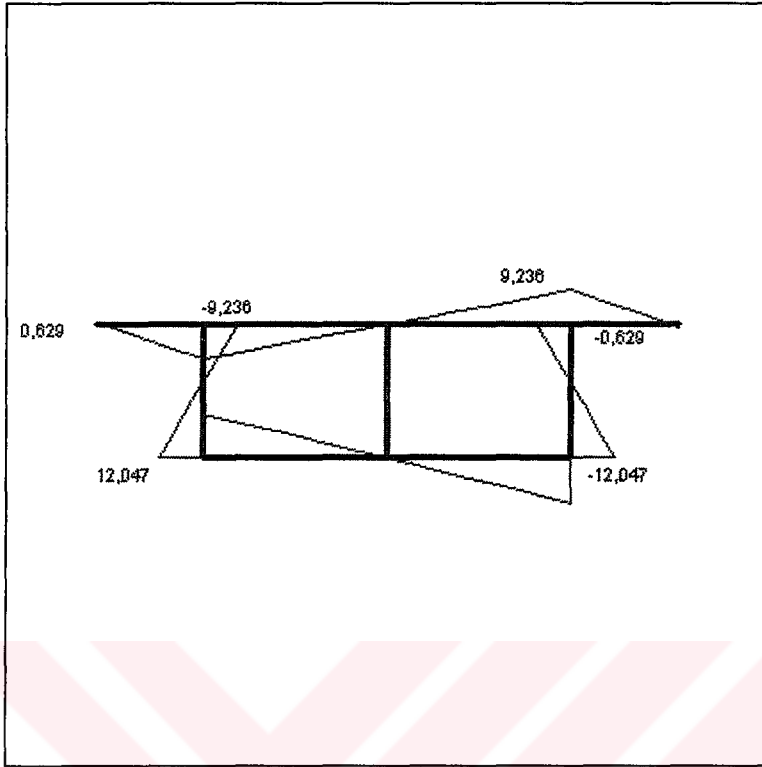


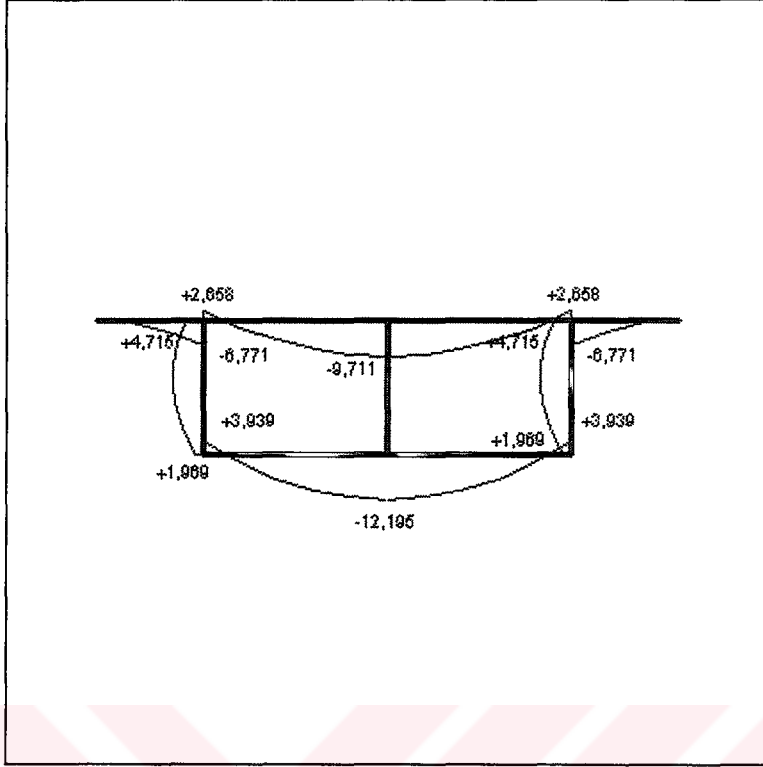
Diyagram 3 - Sekonder Burulma Momenti Diyagramı



Diyagram 4 - Bimoment Diyagramı



**Enkesit Gerilme Diyagramları: 1. Açıklık, Z = 2**



Sekonder Kayma Gerilmesi Diyagramı

## Ek 2 TWS Programı Analiz Kodları

```

Public A() As Double
Public B() As Double
Public C() As Double
Public D() As Double
'Malzeme Özellikleri
Public G As Double      'Kayma modülü
Public E As Double      'Elastisite modülü
'kesit özellikleri
Public ktip As Integer  'Kesit tipi
Public RO As Double     'Açık veya kapalı kesit tanımı ve kapalı kesitle
    ilgili katsayı
Public bust As Double   'Kesit Özellikleri
Public tust As Double   '
Public tgov As Double   '
Public hgov As Double   '
Public balt As Double   '
Public talt As Double   '
Public b1 As Double     '
Public HD() As Double   'Kesit parçalarının dönme merkezine olan
    uzaklıkları
Public pl() As Double   'Kesit parçası uzunlukları
Public pt() As Double   'Kesit parçası kalınlıkları
Public X() As Double    'kesit noktaları X koordinatı
Public Y() As Double    'kesit noktaları Y koordinatı
Public W() As Double    'kesit noktaları W koordinatı
Public kps As Integer   'kesit parça sayısı
Public kns As Integer   'Normal gerilme bulunacak kesit nokta sayısı
Public ktn As Integer   'Kayma gerilmesi bulunacak kesit nokta sayısı
Public F As Double      'Enkesit Alanı
Public FX As Double     'X'e göre statik moment
Public FY As Double     'Y'e göre statik moment
Public FW As Double     'Çarpılma statik momenti
Public XO As Double
Public YO As Double
Public WO As Double
Public FXX As Double    'Sektör
Public FYY As Double    'atalet
Public FWW As Double    'momentleri
Public FXY As Double    'çarpım atalet momenti
Public FWX As Double    'çarpılma çarpım momenti
Public FWY As Double    'çarpılma çarpım momenti
Public tan2fi As Double
Public fi As Double
Public SC(2, 2) As Double
Public YR1 As Double
Public YR2 As Double
Public Dx As Double
Public Dy As Double
Public Dxo As Double
Public Dyo As Double
Public XD As Double
Public YD As Double
Public Jb As Double     'St. Vénant Burulma sabiti
Public CK As Double
Public EK() As Double   'çarpılma yüzeyleri
Public TJ() As Double
Public SWDS() As Double 'Çarpılma Alanları
'Kapalı Kesit -Kesit Özellikleri-
Public SW() As Double   'Kapalı Kesit - Çarpılma Alanı Ek Terim
Public DSBTT As Double  '1/t oranları toplamı

```

Public FIP() As Double '2.Fg/DSBTT - Primer Kayma Gerilmesi Hesabında Kullanılacak

Public FIBT() As Double 'Pri. Kayma Ger. Hes. Noktaların pisi/t oranları

Public R() As Double 'Bur. Sabitinde ek terimde her göz için  
2.Fg/DSBTT

Public Fg() As Double 'Kapalı Kesitte göz Alanı

Public FHH As Double 'Kapalı Kesitte Kısaltma

Public IG As Integer 'Kapalı Kesit - Göz Sayısı

'Çift Gözlü Kapalı Kesit Değişkenleri

Public CSG() As Double 'Göz Çeperi Parça Uzunlukları

Public TG() As Double 'Göz Çeperi Parça Kalınlıkları

Public CSA() As Double 'Gözler Arası Parça Uzunlukları

Public TA() As Double 'Gözler Arası Parça Kalınlıkları

Public DSBTG() As Double 'Göz Çeperi Parça Uzunluk/Kalınlık Oranları  
Toplamı

Public DSBTA As Double 'Gözle Arası Parça Uzunluk/Kalınlık Oranları  
Toplamı

Public FR() As Double 'Çok Gözlü Kapalı Kesit Denklem Takımı Katsayı  
Matrisi

Public FrTers() As Double 'KATSayı Matrisi Ters

Public RI(1 To 2) As Double

Public FWI(1 To 2) As Double

Public SWIK() As Double

'Sistem Çözümü

Public CL() As Double 'Herbir açıklık uzunluğu (L)

Public ZETA() As Double 'Yükün sol mesnetten mesafesi

Public CZ() As Double 'yayıllı burulma momentinin başlangıcının sol  
mesnetten mesafesi

Public aciklik As Integer 'açıklık sayısı

Public YUK() As Double 'yükleme değeri

Public mesnet() As Integer 'Mesnet indeksi

Public Ytip() As Integer 'herbir açıklığın yüklenme tipi

'Geçiş matrisi yöntemi

Public SVB() As Double 'Mesnetten-Mesnete Geçiş matrisi

Public Z() As Double 'Geçiş matrisinin hesaplanacağı çubuk  
mesafeleri

Public Pn As Integer 'Çubuk Değer Hesaplanacak Nokta sayısı

Public PPN As Integer 'Yükten önceki çubuk hesap noktası

Public Pny As Integer 'yayıllı yük için hesap noktaları

Public PPny As Integer 'yayıllı yükün bittiği noktanın solundaki nokta

Public CPS As Integer 'Çubuk Parça Sayısı

Public VB() As Double 'Geçiş Matrisi

Public VBA() As Double 'Geçiş Matrisi, Yükten Sonraki Kısım için

Public SNH() As Double 'sinh(k\*z) -Kesit analizi için

Public CSH() As Double 'cosh(k\*z) -Kesit analizi için

Public SNHA() As Double 'sinh(k\*z) -Kesit analizi için, Yükten Sonraki  
Kısım için

Public CSHA() As Double 'cosh(k\*z) -Kesit analizi için, Yükten Sonraki  
Kısım için

Public VX() As Double 'Başlangıç değerleri matrisi

Public VXX() As Double 'Başlangıç Değerleri Vektörü

Public SVX() As Double 'Başlangıç değerlerinin geçiş matrisi ile  
çarpılma sonucu

Public VP() As Double 'Yük Vektörü

Public VPEK() As Double 'yayıllı burulma momenti için ek yük vektörü

Public SVP() As Double 'Yük vektörünü son mesnete kadar aktaran yük  
matrisi

Public K() As Double 'Süreksizlik matrisi

Public SK() As Double 'Süreksizlik Matrislerinin Taşınmış Durumu

Public V() As Double 'Sonuç Vektörü

```

Public sekme As Integer      'SVB geçiş matrisi sayısı
Public SH() As Double       'sinh(k*z) değeri -Sistem Çözümü için
Public CH() As Double       'cosh(k*z) değeri -Sistem Çözümü için
Public SHA() As Double     'yükten mesnete sinh(k*z) değeri -Sistem Çözümü
    için
Public CHA() As Double     'yükten mesnete cosh(k*z) değeri -Sistem Çözümü
    için
Public SHB() As Double     'Yayıllı burulma momenti için sinh -Sistem
    Çözümü için
Public CHB() As Double     'Yayıllı burulma momenti için cosh -Sistem
    Çözümü için
Public SHC() As Double     'Yayıllı burulma momenti, (sinh) yükün etkidiği
    bölgeden sonra
Public CHC() As Double     'Yayıllı burulma momenti, (cosh) yükün etkidiği
    bölgeden sonra
Public SHEK() As Double    'yayıllı burulma momenti taşınması için ek
Public CHEK() As Double    'yayıllı burulma momenti taşınması için ek
Public SVBEK() As Double   'Yayıllı burulma momenti için ek taşıma marisi
'Denklem Çözümü Değişkenleri
Public KM() As Double     'Denklem Takımının Katsayılar Matrisi
Public KP() As Double     'Denklem Takımının Karşı Tarafı
Public KMTERS()          'Katsayılar matrisinin tersi
Public KTERS()           'Matris Tersini alma ara işlemler için
Public MTERS()           'Matris Tersini alma ara işlemler için
Public ARACAR() As Double 'Matris Tersini alma ara işlemler için
'Bilinmeyenlerin Bulunması
Public BILBUL() As Double 'Bilinmeyenlerin Yer Bulunduğu Vektör
'Kesit Tesirleri ve Gerilmeler
Public MBS() As Double
Public MBP() As Double
Public MB() As Double
Public BM() As Double
Public FIM() As Double    'Z ekseninde Kesit Dönme Açısı
Public SIGMA() As Double
Public TOPP() As Double
Public TOSK() As Double
'Analiz Tamamlandı
Public analiz As Boolean
Public kesitvar As Boolean
'sistem Çözümü
Public Solution_Problem As Boolean

```

```
Public Sub main()
```

```
    On Local Error GoTo sistemhata
```

```
    'Analizin Başlatılabilmesi için kontrollerin yapılması
```

```
    If aciklik = 0 Then
```

```
        cevap = MsgBox("Sistem Tanımlanmamış!" & Chr(13) & "Analiz
        Yapılamaz", vbExclamation, "Analiz Başlatılmıyor...")
```

```
        Exit Sub
```

```
    End If
```

```
    If mesvar = False Then
```

```
        cevap = MsgBox("Mesnet Şartları Tanımlanmamış!" & Chr(13) & "Analiz
        Yapılamaz", vbExclamation, "Analiz Başlatılmıyor...")
```

```
        Exit Sub
```

```
    End If
```

```
    If kesitvar = False Then
```

```
        cevap = MsgBox("Enkesit Tanımlanmamış!" & Chr(13) & "Analiz
        Yapılamaz", vbExclamation, "Analiz Başlatılmıyor...")
```

```

Exit Sub
End If
If yukvar = False Then
    cevap = MsgBox("Yükleme Tanımlanmamış!" & Chr(13) & "Analiz
        Yapılamaz", vbExclamation, "Analiz Başlatılamıyor...")
    Exit Sub
End If

'*** ANALİZİN BAŞLATILMASI ***

Call Kesit_Hesapla
Call asal_eksen
Call VX_MATRIS
Call SISTEM
Call VP_yuk_matrisi
If aciklik = 1 Then
    Call Sistem_Cozumu_Tek_Aciklik
Else
    Call K_mesnet_matrisi
    Call DENKLEMLER
    Call sistem_cozum
    Call Matris_Inverse
    Call bilinmeyen_bul
    Call Kesit_Tesirleri
End If
'Enkesit Üzerinde Gerilmeleri Hesplanışı
Call carpilma
Call kesit_gerilme

'SİSTEM DAVRANIŞLARININ AYARLANMASI

'Analiz İşleminin Tamamlanması
analiz = True

'Sonuç ToolBar'ının Açılması
frmMain.MainToolBar.Bands("TBarSonuc").Visible = True
'Status Bar Mesajı
frmMain.MainToolBar.Bands("StatusBar").Tools("StDurum").Caption =
    "Analiz Tamamlandı!"

Call frmMain.kilitle

MsgBox "Analiz Tamamlandı...!"

Exit Sub

sistemhata:
    cevap = MsgBox("Analiz Tamamlanamadı!", vbExclamation)

End Sub

```

```

Public Sub Kesit_Hesapla()
  'Kesit Özellikleri
  Select Case ktip
    Case 1: 'I Kesit ise
      kps = 24
      RO = 1
      kns = 25
      ktn = 29
    Case 2: 'C Kesit Tipi Seçili Olma Durumu
      kps = 16
      RO = 1
      kns = 17
      ktn = 19
    Case 3: 'Z Kesit Tipi Seçili Olma Durumu
      kps = 16
      RO = 1
      kns = 17
      ktn = 19
    Case 4: 'Tek Gözlü Kapalı Kesit Olması Durumu
      kps = 38
      RO = 0
      kns = kps
      ktn = 44
      IG = 1
    Case 5: 'Çift Gözlü Kapalı Kesit Olması Durumu
      kps = 44
      RO = 0
      kns = 43
      ktn = 53
      IG = 2
  End Select

  ReDim A(1 To 3, 1 To kps)
  ReDim B(1 To 3, 1 To kps)
  ReDim C(1 To 3, 1 To kps)
  ReDim D(1 To 3, 1 To kps)
  ReDim pt(1 To kps)
  ReDim pl(1 To kps)
  ReDim HD(1 To kps)

  'Kesit Parçalarının Başlangıç ve Bitiş Noktalarının Koordinatlarının
  Hesaplanması
  If ktip = 1 Then
    '-----
    '***          I KESİT          ***
    '-----

    'Başlangıç Noktalarının X koordinatları
    For i = 1 To kps
      If i < 7 Then A(1, i) = 0
      If i = 7 Then A(1, i) = A(1, i - 2) + bust / 8
      If i = 8 Then A(1, i) = -A(1, i - 1)
      If i = 9 Then A(1, i) = A(1, i - 2) + bust / 8
      If i = 10 Then A(1, i) = -A(1, i - 1)
      If i = 11 Then A(1, i) = A(1, i - 2) + bust / 8
      If i = 12 Then A(1, i) = -A(1, i - 1)
      If i >= 13 And i <= 18 Then A(1, i) = 0
      If i = 19 Then A(1, i) = A(1, i - 2) + balt / 8
      If i = 20 Then A(1, i) = -A(1, i - 1)
      If i = 21 Then A(1, i) = A(1, i - 2) + balt / 8
      If i = 22 Then A(1, i) = -A(1, i - 1)
      If i = 23 Then A(1, i) = A(1, i - 2) + balt / 8
    Next i
  End If

```

```

    If i = 24 Then A(1, i) = -A(1, i - 1)
Next
'Başlangıç Noktalarının Y koordinatları
For i = 1 To kps
    If i = 1 Then A(2, i) = 0
    If i >= 2 And i <= 5 Then A(2, i) = A(2, i - 1) - hgov / 8
    If i >= 6 And i <= 12 Then A(2, i) = A(2, i - 1)
    If i = 13 Then A(2, i) = A(2, 1)
    If i >= 14 And i <= 17 Then A(2, i) = A(2, i - 1) + hgov / 8
    If i > 17 Then A(2, i) = A(2, i - 1)
Next
'Bitiş Noktalarının X koordinatları
For i = 1 To kps
    If i <= 4 Then B(1, i) = 0
    If i = 5 Then B(1, i) = B(1, i - 1) + bust / 8
    If i = 6 Then B(1, i) = -B(1, i - 1)
    If i = 7 Then B(1, i) = B(1, i - 2) + bust / 8
    If i = 8 Then B(1, i) = -B(1, i - 1)
    If i = 9 Then B(1, i) = B(1, i - 2) + bust / 8
    If i = 10 Then B(1, i) = -B(1, i - 1)
    If i = 11 Then B(1, i) = B(1, i - 2) + bust / 8
    If i = 12 Then B(1, i) = -B(1, i - 1)
    If i >= 13 And i <= 16 Then B(1, i) = 0
    If i = 17 Then B(1, i) = B(1, i - 1) + balt / 8
    If i = 18 Then B(1, i) = -B(1, i - 1)
    If i = 19 Then B(1, i) = B(1, i - 2) + balt / 8
    If i = 20 Then B(1, i) = -B(1, i - 1)
    If i = 21 Then B(1, i) = B(1, i - 2) + balt / 8
    If i = 22 Then B(1, i) = -B(1, i - 1)
    If i = 23 Then B(1, i) = B(1, i - 2) + balt / 8
    If i = 24 Then B(1, i) = -B(1, i - 1)
Next
'Bitiş Noktalarının Y koordinatları
For i = 1 To kps
    If i = 1 Then B(2, i) = -hgov / 8
    If i >= 2 And i <= 4 Then B(2, i) = B(2, i - 1) - hgov / 8
    If i >= 5 And i <= 12 Then B(2, i) = B(2, i - 1)
    If i = 13 Then B(2, i) = A(2, 1) + hgov / 8
    If i >= 14 And i <= 16 Then B(2, i) = B(2, i - 1) + hgov / 8
    If i >= 17 And i <= 24 Then B(2, i) = B(2, i - 1)
Next
'Kesit parçaları uzunluk ve kalınlıkları
For i = 1 To kps
    If i < 5 Then
        pt(i) = tgov
        pl(i) = hgov / 8
    End If
    If i >= 5 And i <= 12 Then
        pt(i) = tust
        pl(i) = bust / 8
    End If
    If i >= 13 And i <= 16 Then
        pt(i) = tgov
        pl(i) = hgov / 8
    End If
    If i >= 17 Then
        pt(i) = talt
        pl(i) = balt / 8
    End If
Next
'Kesit Parçalarının Dönme Merkezine Dik Mesafeleri
For i = 1 To kps

```

```

HD(i) = Abs((A(1, i) - B(1, i)) * A(2, i) - (A(2, i) - B(2, i)) *
A(1, i)) / (Sqr((A(1, i) - B(1, i)) ^ 2 + (A(2, i) - B(2, i)) ^
2)))
Next
'Başlangıç Noktalarının birim çarpılma koordinatları
For i = 1 To kps
  If i < 5 Then A(3, i) = 0
  If i = 5 Then A(3, i) = HD(i) * Abs(A(1, i))
  If i = 6 Then A(3, i) = (-1) * A(3, i - 1)
  If i = 7 Then A(3, i) = HD(i) * Abs(A(1, i))
  If i = 8 Then A(3, i) = (-1) * A(3, i - 1)
  If i = 9 Then A(3, i) = HD(i) * Abs(A(1, i))
  If i = 10 Then A(3, i) = (-1) * A(3, i - 1)
  If i = 11 Then A(3, i) = HD(i) * Abs(A(1, i))
  If i = 12 Then A(3, i) = (-1) * A(3, i - 1)
  If i >= 13 And i <= 16 Then A(3, i) = 0
  If i = 17 Then A(3, i) = (-1) * HD(i) * Abs(A(1, i))
  If i = 18 Then A(3, i) = (-1) * A(3, i - 1)
  If i = 19 Then A(3, i) = (-1) * HD(i) * Abs(A(1, i))
  If i = 20 Then A(3, i) = (-1) * A(3, i - 1)
  If i = 21 Then A(3, i) = (-1) * HD(i) * Abs(A(1, i))
  If i = 22 Then A(3, i) = (-1) * A(3, i - 1)
  If i = 23 Then A(3, i) = (-1) * HD(i) * Abs(A(1, i))
  If i = 24 Then A(3, i) = (-1) * A(3, i - 1)
Next
'Bitiş Noktalarının birim çarpılma koordinatları
For i = 1 To kps
  If i <= 4 Then B(3, i) = 0
  If i = 5 Then B(3, i) = HD(i) * Abs(B(1, i))
  If i = 6 Then B(3, i) = (-1) * B(3, i - 1)
  If i = 7 Then B(3, i) = HD(i) * Abs(B(1, i))
  If i = 8 Then B(3, i) = (-1) * B(3, i - 1)
  If i = 9 Then B(3, i) = HD(i) * Abs(B(1, i))
  If i = 10 Then B(3, i) = (-1) * B(3, i - 1)
  If i = 11 Then B(3, i) = HD(i) * Abs(B(1, i))
  If i = 12 Then B(3, i) = (-1) * B(3, i - 1)
  If i >= 13 And i <= 16 Then B(3, i) = 0
  If i = 17 Then B(3, i) = (-1) * HD(i) * Abs(B(1, i))
  If i = 18 Then B(3, i) = (-1) * B(3, i - 1)
  If i = 19 Then B(3, i) = (-1) * HD(i) * Abs(B(1, i))
  If i = 20 Then B(3, i) = (-1) * B(3, i - 1)
  If i = 21 Then B(3, i) = (-1) * HD(i) * Abs(B(1, i))
  If i = 22 Then B(3, i) = (-1) * B(3, i - 1)
  If i = 23 Then B(3, i) = (-1) * HD(i) * Abs(B(1, i))
  If i = 24 Then B(3, i) = (-1) * B(3, i - 1)
Next
ElseIf ktip = 2 Then
'-----
'***          C KESİT          ***
'-----
'Başlangıç Noktalarının X koordinatları
For i = 1 To kps
  If i <= 5 Then A(1, i) = 0
  If i >= 6 And i <= 8 Then A(1, i) = A(1, i - 1) - bust / 4
  If i >= 9 And i <= 13 Then A(1, i) = 0
  If i >= 14 And i <= 16 Then A(1, i) = A(1, i - 1) - balt / 4
Next
'Başlangıç Noktalarının Y koordinatları
For i = 1 To kps
  If i = 1 Then A(2, i) = 0
  If i >= 2 And i <= 5 Then A(2, i) = A(2, i - 1) - hgov / 8
  If i >= 6 And i <= 8 Then A(2, i) = A(2, i - 1)

```

```

    If i = 9 Then A(2, i) = 0
    If i > 9 And i <= 13 Then A(2, i) = A(2, i - 1) + hgov / 8
    If i >= 14 And i <= 16 Then A(2, i) = A(2, i - 1)
Next
'Bitiş Noktalarının X koordinatları
For i = 1 To kps
    If i <= 4 Then B(1, i) = 0
    If i >= 5 And i <= 8 Then B(1, i) = B(1, i - 1) - bust / 4
    If i >= 9 And i <= 12 Then B(1, i) = 0
    If i >= 13 And i <= 16 Then B(1, i) = B(1, i - 1) - balt / 4
Next
'Bitiş Noktalarının Y koordinatları
For i = 1 To kps
    If i = 1 Then B(2, i) = -hgov / 8
    If i >= 2 And i <= 4 Then B(2, i) = B(2, i - 1) - hgov / 8
    If i >= 5 And i <= 8 Then B(2, i) = B(2, i - 1)
    If i = 9 Then B(2, i) = hgov / 8
    If i >= 10 And i <= 12 Then B(2, i) = B(2, i - 1) + hgov / 8
    If i >= 13 And i <= 16 Then B(2, i) = B(2, i - 1)
Next
'Kesit parçaları uzunluk ve kalınlıkları
For i = 1 To kps
    'Et Kalınlıkları
    If i >= 1 And i <= 4 Then pt(i) = tgov
    If i >= 5 And i <= 8 Then pt(i) = tust
    If i >= 9 And i <= 12 Then pt(i) = tgov
    If i >= 13 And i <= 16 Then pt(i) = talt
    'Kesit Parça Uzunlukları
    If i >= 1 And i <= 4 Then pl(i) = hgov / 8
    If i >= 5 And i <= 8 Then pl(i) = bust / 4
    If i >= 9 And i <= 12 Then pl(i) = hgov / 8
    If i >= 13 And i <= 16 Then pl(i) = balt / 4
Next
'Kesit Parçalarının Dönme Merkezine Dik Mesafeleri
For i = 1 To kps
    HD(i) = Abs(((A(1, i) - B(1, i)) * A(2, i) - (A(2, i) - B(2, i)) *
        A(1, i)) / (Sqr((A(1, i) - B(1, i)) ^ 2 + (A(2, i) - B(2, i)) ^
        2)))
Next
'Başlangıç Noktalarının birim çarpılma koordinatları
For i = 1 To kps
    If i >= 1 And i <= 4 Then A(3, i) = HD(i) * Abs(A(2, i))
    If i >= 5 And i <= 8 Then A(3, i) = -HD(i) * Abs(A(1, i))
    If i >= 9 And i <= 12 Then A(3, i) = HD(i) * Abs(A(2, i))
    If i >= 13 And i <= 16 Then A(3, i) = HD(i) * Abs(A(1, i))
Next
'Bitiş Noktalarının birim çarpılma koordinatları
For i = 1 To kps
    If i >= 1 And i <= 4 Then B(3, i) = HD(i) * Abs(B(2, i))
    If i >= 5 And i <= 8 Then B(3, i) = -HD(i) * Abs(B(1, i))
    If i >= 9 And i <= 12 Then B(3, i) = HD(i) * Abs(B(2, i))
    If i >= 13 And i <= 16 Then B(3, i) = HD(i) * Abs(B(1, i))
Next
ElseIf ktıp = 3 Then
    '-----
    '***      Z KESİT      ***
    '-----

'Başlangıç Noktalarının A(1,i) - X koordinatları
For i = 1 To kps
    If i <= 5 Then A(1, i) = 0
    If i >= 6 And i <= 8 Then A(1, i) = A(1, i - 1) - bust / 4

```

```

    If i >= 9 And i <= 13 Then A(1, i) = 0
    If i >= 14 And i <= 16 Then A(1, i) = A(1, i - 1) + balt / 4
Next
'Başlangıç Noktalarının A(2,i) - Y koordinatları
For i = 1 To kps
    If i = 1 Then A(2, i) = 0
    If i >= 2 And i <= 5 Then A(2, i) = A(2, i - 1) - hgov / 8
    If i >= 6 And i <= 8 Then A(2, i) = A(2, i - 1)
    If i = 9 Then A(2, i) = 0
    If i >= 10 And i <= 13 Then A(2, i) = A(2, i - 1) + hgov / 8
    If i >= 14 Then A(2, i) = A(2, i - 1)
Next
'Bitiş Noktalarının B(1,i) - X koordinatları
For i = 1 To kps
    If i >= 1 And i <= 4 Then B(1, i) = 0
    If i >= 5 And i <= 8 Then B(1, i) = B(1, i - 1) - bust / 4
    If i >= 9 And i <= 12 Then B(1, i) = 0
    If i >= 13 And i <= 16 Then B(1, i) = B(1, i - 1) + balt / 4
Next
'Bitiş Noktalarının B(2,i) - Y koordinatları
For i = 1 To kps
    If i = 1 Then B(2, i) = -hgov / 8
    If i >= 2 And i <= 4 Then B(2, i) = B(2, i - 1) - hgov / 8
    If i >= 5 And i <= 8 Then B(2, i) = B(2, i - 1)
    If i = 9 Then B(2, i) = hgov / 8
    If i >= 10 And i <= 12 Then B(2, i) = B(2, i - 1) + hgov / 8
    If i >= 13 And i <= 16 Then B(2, i) = B(2, i - 1)
Next
'Kesit parçaları uzunluk ve kalınlıkları
For i = 1 To kps
    'Et Kalınlıkları
    If i >= 1 And i <= 4 Then pt(i) = tgov
    If i >= 5 And i <= 8 Then pt(i) = tust
    If i >= 9 And i <= 12 Then pt(i) = tgov
    If i >= 13 Then pt(i) = talt
    'Kesit Parça Uzunlukları
    If i >= 1 And i <= 4 Then pl(i) = hgov / 8
    If i >= 5 And i <= 8 Then pl(i) = bust / 4
    If i >= 9 And i <= 12 Then pl(i) = hgov / 8
    If i >= 13 Then pl(i) = balt / 4
Next
'Kesit Parçalarının Dönme Merkezine Dik Mesafeleri
For i = 1 To kps
    HD(i) = Abs((A(1, i) - B(1, i)) * A(2, i) - (A(2, i) - B(2, i)) *
        A(1, i)) / (Sqr((A(1, i) - B(1, i)) ^ 2 + (A(2, i) - B(2, i)) ^
        2)))
Next
'Başlangıç Noktalarının birim çarpılma koordinatları
For i = 1 To kps
    If i >= 1 And i <= 4 Then A(3, i) = HD(i) * Abs(A(2, i))
    If i >= 5 And i <= 8 Then A(3, i) = -HD(i) * Abs(A(1, i))
    If i >= 9 And i <= 12 Then A(3, i) = HD(i) * Abs(A(2, i))
    If i >= 13 And i <= 16 Then A(3, i) = -HD(i) * Abs(A(1, i))
Next
'Bitiş Noktalarının birim çarpılma koordinatları
For i = 1 To kps
    If i >= 1 And i <= 4 Then B(3, i) = HD(i) * Abs(B(2, i))
    If i >= 5 And i <= 8 Then B(3, i) = -HD(i) * Abs(B(1, i))
    If i >= 9 And i <= 12 Then B(3, i) = HD(i) * Abs(B(2, i))
    If i >= 13 And i <= 16 Then B(3, i) = -HD(i) * Abs(B(1, i))
Next

```

```

ElseIf ktip = 4 Then
'-----
'***      TEK GÖZLÜ KAPALI KESİT      ***
'-----

'Başlangıç Noktalarının A(1,i) - X koordinatları
For i = 1 To kps
  If i = 1 Then A(1, i) = 0
  If i >= 2 And i <= 6 Then A(1, i) = A(1, i - 1) + (balt / 10)
  If i >= 7 And i <= 12 Then A(1, i) = A(1, i - 1)
  If i >= 13 And i <= 16 Then A(1, i) = A(1, i - 1) - (balt / 10)
  If i = 17 Then A(1, i) = A(1, 6) + b1 / 3
  If i = 18 Or i = 19 Then A(1, i) = A(1, i - 1) + b1 / 3
  If i > 19 Then A(1, i) = -A(1, i - 19)
Next
'Başlangıç Noktalarının A(2,i) - Y koordinatları
For i = 1 To kps
  If i <= 6 Then A(2, i) = 0
  If i >= 7 And i <= 12 Then A(2, i) = A(2, i - 1) + hgov / 6
  If i >= 13 And i <= 16 Then A(2, i) = A(2, i - 1)
  If i >= 17 And i <= 19 Then A(2, i) = 0
  If i > 19 Then A(2, i) = A(2, i - 19)
Next
'Bitiş Noktalarının X koordinatları
For i = 1 To kps
  If i = 1 Then B(1, i) = (balt / 10)
  If i >= 2 And i <= 5 Then B(1, i) = B(1, i - 1) + (balt / 10)
  If i >= 6 And i <= 11 Then B(1, i) = B(1, i - 1)
  If i >= 12 And i <= 16 Then B(1, i) = B(1, i - 1) - (balt / 10)
  If i = 17 Then B(1, i) = B(1, 5)
  If i >= 18 And i <= 19 Then B(1, i) = B(1, i - 1) + (b1 / 3)
  If i > 19 Then B(1, i) = -B(1, i - 19)
Next
'Bitiş Noktalarının Y koordinatları
For i = 1 To kps
  If i >= 1 And i <= 5 Then B(2, i) = 0
  If i = 6 Then B(2, i) = hgov / 6
  If i >= 7 And i <= 11 Then B(2, i) = B(2, i - 1) + hgov / 6
  If i >= 12 And i <= 16 Then B(2, i) = B(2, i - 1)
  If i >= 17 And i <= 19 Then B(2, i) = 0
  If i > 19 Then B(2, i) = B(2, i - 19)
Next
'Kesit parçaları uzunluk ve kalınlıkları
For i = 1 To kps
  'Et Kalınlıkları
  If i >= 1 And i <= 5 Then pt(i) = tust
  If i >= 6 And i <= 11 Then pt(i) = tgov
  If i >= 12 And i <= 16 Then pt(i) = talt
  If i >= 17 And i <= 19 Then pt(i) = tust
  If i > 19 Then pt(i) = pt(i - 19)
  'Kesit Parça Uzunlukları
  If i >= 1 And i <= 5 Then pl(i) = (balt / 10)
  If i >= 6 And i <= 11 Then pl(i) = (hgov / 6)
  If i >= 12 And i <= 16 Then pl(i) = (balt / 10)
  If i >= 17 And i <= 19 Then pl(i) = (b1 / 3)
  If i > 19 Then pl(i) = pl(i - 19)
Next

ReDim Fg(1 To 1)
'GÖZ ALANI
Fg(1) = balt * hgov

```

'w - Birim çarpılma değerlerinin ayarlanması  
Call w\_ilk\_kapali

ElseIf ktip = 5 Then

```

'-----
'****      ÇİFT GÖZLÜ KAPALI KESİT      ****
'-----
'Başlangıç Noktalarının A(1,i) - X koordinatları
For i = 1 To kps
  If i = 1 Then A(1, i) = 0
  If i >= 2 And i <= 6 Then A(1, i) = A(1, i - 1) + (balt / 10)
  If i >= 7 And i <= 12 Then A(1, i) = A(1, i - 1)
  If i >= 13 And i <= 17 Then A(1, i) = A(1, i - 1) - (balt / 10)
  If i >= 18 And i <= 22 Then A(1, i) = 0
  If i = 23 Then A(1, i) = A(1, 6) + b1 / 3
  If i = 24 Or i = 25 Then A(1, i) = A(1, i - 1) + b1 / 3
  If i > 25 And i < 42 Then A(1, i) = -A(1, i - 25)
  If i >= 42 Then A(1, i) = -A(1, i - 19)
Next
'Başlangıç Noktalarının A(2,i) - Y koordinatları
For i = 1 To kps
  If i <= 6 Then A(2, i) = 0
  If i >= 7 And i <= 12 Then A(2, i) = A(2, i - 1) + hgov / 6
  If i >= 13 And i <= 17 Then A(2, i) = A(2, i - 1)
  If i >= 18 And i <= 22 Then A(2, i) = A(2, i - 1) - hgov / 6
  If i >= 23 And i <= 25 Then A(2, i) = 0
  If i > 25 And i < 42 Then A(2, i) = A(2, i - 25)
  If i >= 42 Then A(2, i) = A(2, i - 19)
Next
'Bitiş Noktalarının X koordinatları
For i = 1 To kps
  If i = 1 Then B(1, i) = (balt / 10)
  If i >= 2 And i <= 5 Then B(1, i) = B(1, i - 1) + (balt / 10)
  If i >= 6 And i <= 11 Then B(1, i) = B(1, i - 1)
  If i >= 12 And i <= 16 Then B(1, i) = B(1, i - 1) - (balt / 10)
  If i >= 17 And i <= 22 Then B(1, i) = B(1, i - 1)
  If i = 23 Then B(1, i) = B(1, 5)
  If i = 24 Or i = 25 Then B(1, i) = B(1, i - 1) + (b1 / 3)
  If i > 25 And i < 42 Then B(1, i) = -B(1, i - 25)
  If i >= 42 Then B(1, i) = -B(1, i - 19)
Next
'Bitiş Noktalarının Y koordinatları
For i = 1 To kps
  If i >= 1 And i <= 5 Then B(2, i) = 0
  If i = 6 Then B(2, i) = hgov / 6
  If i >= 7 And i <= 11 Then B(2, i) = B(2, i - 1) + hgov / 6
  If i >= 12 And i <= 16 Then B(2, i) = B(2, i - 1)
  If i >= 17 And i <= 22 Then B(2, i) = B(2, i - 1) - hgov / 6
  If i >= 23 And i <= 25 Then B(2, i) = 0
  If i > 25 And i < 42 Then B(2, i) = B(2, i - 25)
  If i >= 42 Then B(2, i) = B(2, i - 19)
Next
'Kesit parçaları uzunluk ve kalınlıkları
For i = 1 To kps
  'Et Kalınlıkları
  If i >= 1 And i <= 5 Then pt(i) = tust
  If i >= 6 And i <= 11 Then pt(i) = tgov
  If i >= 12 And i <= 16 Then pt(i) = talt
  If i >= 17 And i <= 22 Then pt(i) = tgov
  If i >= 23 And i <= 25 Then pt(i) = tust
  If i > 25 And i < 42 Then pt(i) = pt(i - 25)
  If i >= 42 Then pt(i) = pt(i - 19)

```

```

'Kesit Parça Uzunlukları
If i >= 1 And i <= 5 Then pl(i) = balt / 10
If i >= 6 And i <= 11 Then pl(i) = hgov / 6
If i >= 12 And i <= 16 Then pl(i) = balt / 10
If i >= 17 And i <= 22 Then pl(i) = hgov / 6
If i >= 23 And i <= 25 Then pl(i) = b1 / 3
If i > 25 And i < 42 Then pl(i) = pl(i - 25)
If i >= 42 Then pl(i) = pl(i - 19)
Next
ReDim Fg(1 To 2)
'GÖZ ALANI
Fg(1) = balt / 2 * hgov
Fg(2) = balt / 2 * hgov
'w - Birim çarpılma değerlerinin ayarlanması
Call kapali_kesit_on_hesap
Call w_ilk_kapali

End If      'Kesit Tipi Seçimi I,C,Z,K1,K2 Kesit Tipleri Kesit
           Özellikleri Hesaplanması

'Enkesit Alanı
F = 0
For i = 1 To kps
    F = F + pl(i) * pt(i)
Next

'Kesit noktaları X,Y,W koordinatlarının bulunması

If ktip = 1 Then
    '*** I Kesit için ***
    ReDim X(1 To kns)
    ReDim Y(1 To kns)
    ReDim W(1 To kns)

    For i = 1 To kns
        If i <= 5 Then
            X(i) = A(1, i)
            Y(i) = A(2, i)
            W(i) = A(3, i)
        Else
            X(i) = B(1, i - 1)
            Y(i) = B(2, i - 1)
            W(i) = B(3, i - 1)
        End If
    Next
ElseIf ktip = 2 Then
    '*** C Kesit için ***
    ReDim X(1 To kns)
    ReDim Y(1 To kns)
    ReDim W(1 To kns)
    For i = 1 To kns
        If i = 9 Or i = 17 Then
            X(i) = B(1, i - 1)
            Y(i) = B(2, i - 1)
            W(i) = B(3, i - 1)
        Else
            X(i) = A(1, i)
            Y(i) = A(2, i)
            W(i) = A(3, i)
        End If
    Next
Next

```

```

ElseIf ktip = 3 Then
  '*** Z Kesit için ***
  ReDim X(1 To kns)
  ReDim Y(1 To kns)
  ReDim W(1 To kns)
  For i = 1 To kns
    If i = 9 Or i = 17 Then
      X(i) = B(1, i - 1)
      Y(i) = B(2, i - 1)
      W(i) = B(3, i - 1)
    Else
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    End If
  Next
ElseIf ktip = 4 Then
  '*** Tek Gözlü Kapalı Kesit için ***
  ReDim X(1 To kns)
  ReDim Y(1 To kns)
  ReDim W(1 To kns)
  For i = 1 To kns
    If i >= 1 And i <= 16 Then
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    ElseIf i = 17 Then
      X(i) = B(1, i - 1)
      Y(i) = B(2, i - 1)
      W(i) = B(3, i - 1)
    ElseIf i >= 18 And i <= 20 Then
      X(i) = A(1, i - 1)
      Y(i) = A(2, i - 1)
      W(i) = A(3, i - 1)
    ElseIf i >= 21 Then
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    End If
  Next
ElseIf ktip = 5 Then
  '*** Çift Gözlü Kapalı Kesit ***
  ReDim X(1 To kns)
  ReDim Y(1 To kns)
  ReDim W(1 To kns)
  For i = 1 To kns
    If i >= 1 And i <= 25 Then
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    ElseIf i >= 26 And i <= 40 Then
      X(i) = B(1, i)
      Y(i) = B(2, i)
      W(i) = B(3, i)
    ElseIf i >= 41 And i <= 43 Then
      X(i) = A(1, i + 1)
      Y(i) = A(2, i + 1)
      W(i) = A(3, i + 1)
    End If
  Next
End If
End Sub

```

```

Public Sub kapali_kesit_on_hesap()
    ReDim CSG(1 To 2, 1 To 22)
    ReDim TG(1 To 2, 1 To 22)
    ReDim CSA(1 To 1, 1 To 6)
    ReDim TA(1 To 1, 1 To 6)
    ReDim DSBTG(1 To 2)
    ReDim FR(1 To 2, 1 To 2)
    ReDim R(1 To 2)

    'Gözlerin parça uzunlukları ve parça kalınlıklarının belirlenmesi
    For i = 1 To IG
        For j = 1 To 22
            CSG(i, j) = pl(j)
            TG(i, j) = pt(j)
        Next
    Next
    'Gözlerin Ara Çeperlerin parça uzunlukları ve kalınlıklarının belirlenmesi
    For i = 1 To 1
        For j = 1 To 6
            CSA(i, j) = pl(j + 16)
            TA(i, j) = pt(j + 16)
        Next
    Next
    'Gözler için parça uzunluk / kalınlıkları oranlarının toplamının belirlenmesi
    For i = 1 To IG
        DSBTG(i) = 0
        For j = 1 To 22
            DSBTG(i) = DSBTG(i) + CSG(i, j) / TG(i, j)
        Next
    Next
    DSBTA = 0
    For j = 1 To 6
        DSBTA = DSBTA + CSA(1, j) / TA(1, j)
    Next
    For i = 1 To IG
        FR(i, i) = DSBTG(i)
        If i = 2 Then
            FR(i - 1, i) = -DSBTA
            FR(i, i - 1) = -DSBTA
        End If
    Next
    For i = 1 To IG
        R(i) = 2 * Fg(i)
    Next
    Call denklem_cozum
End Sub

```

```

Public Sub denklem_cozum()
    'Katsayılar matrisi KM nin tersinin alınması
    Dim AraMatris() As Double
    Dim MatTersA() As Double
    Dim FrTersHes() As Double
    Dim ElemanSayisi As Integer

    ElemanSayisi = 1
    ReDim AraMatris(ElemanSayisi + 1, ElemanSayisi, ElemanSayisi)
    ReDim MatTersA(ElemanSayisi + 1, ElemanSayisi, ElemanSayisi)
    ReDim FrTersHes(ElemanSayisi, ElemanSayisi, ElemanSayisi)
    ReDim FrTers(ElemanSayisi, ElemanSayisi)

```

```

'ReDim KMTTERS(aciklik, aciklik)           = FrTers
'ReDim MTTERS(aciklik + 1, aciklik, aciklik) = MatTersA
'ReDim KTTERS(aciklik, aciklik, aciklik)   = FrTersHes

For i = 0 To ElemanSayisi
  For j = 0 To ElemanSayisi
    AraMatris(0, i, j) = FR(i + 1, j + 1)
  Next
Next
For i = 0 To ElemanSayisi
  For j = 0 To ElemanSayisi
    MatTersA(i, j, i) = -AraMatris(i, j, i)
  Next
  n = 0
  Do While n <= ElemanSayisi
    MatTersA(i, n, n) = AraMatris(i, i, i)
    n = n + 1
  Loop
  MatTersA(i, i, i) = 1
  For j = 0 To ElemanSayisi
    For n = 0 To ElemanSayisi
      For m = 0 To ElemanSayisi
        AraMatris(i + 1, j, n) = AraMatris(i + 1, j, n) + MatTersA(i,
          j, m) * AraMatris(i, m, n)
      Next
    Next
  Next
Next
Next 'Denklemin sonu
'Son adımda elde edilen matrisinin tersinin bulunması
For i = 0 To ElemanSayisi
  MatTersA(ElemanSayisi + 1, i, i) = 1 / AraMatris(ElemanSayisi + 1, i,
    i)
Next
'KM Matrisinin
'A2*A1
For i = 0 To ElemanSayisi
  For n = 0 To ElemanSayisi
    For j = 0 To ElemanSayisi
      FrTersHes(0, i, n) = FrTersHes(0, i, n) + MatTersA(1, i, j) *
        MatTersA(0, j, n)
    Next
  Next
Next
Next
'Diğer Çarpımlar
For l = 1 To ElemanSayisi 'MTTERS matrisinden bir düşük satırda
  For n = 0 To ElemanSayisi
    For i = 0 To ElemanSayisi
      For j = 0 To ElemanSayisi
        FrTersHes(l, i, n) = FrTersHes(l, i, n) + MatTersA(l + 1, i, j) *
          FrTersHes(l - 1, j, n)
      Next
    Next
  Next
Next
Next
'Ters Matrisin KMTTERS matrisine aktarılması
For i = 0 To ElemanSayisi
  For j = 0 To ElemanSayisi
    FrTers(i, j) = FrTersHes(ElemanSayisi, i, j)
  Next
Next

```

```

'RI çözüm Vektörünün Hesaplanması
For i = 0 To ElemanSayisi
  RI(i + 1) = 0
  For j = 0 To ElemanSayisi
    RI(i + 1) = RI(i + 1) + FrTers(i, j) * R(j + 1)
  Next
Next
End Sub

```

```

Public Sub w_ilk_kapali()
  ReDim FIP(1 To kps)
  ReDim FIBT(1 To ktn)
  If ktip = 4 Then
    '-----
    '***   TEK GÖZLÜ KAPALI KESİT   ***
    '-----
    DSBTT = 0
    For i = 1 To 16
      DSBTT = DSBTT + 2 * pl(i) / pt(i)
    Next
    For i = 1 To kps
      FIP(i) = 2 * Fg(1) / DSBTT
      If i >= 17 And i <= 19 Then
        FIP(i) = 0
      ElseIf i >= 36 And i <= 38 Then
        FIP(i) = 0
      End If
    Next
    'Kesit Parçalarının Dönme Merkezine Dik Mesafeleri
    For i = 1 To kps
      HD(i) = Abs(((A(1, i) - B(1, i)) * A(2, i) - (A(2, i) - B(2,
        i)) * A(1, i)) / (Sqr((A(1, i) - B(1, i)) ^ 2 + (A(2, i) - B(2,
        i)) ^ 2)))
    Next
    'w koordinatlarının hesaplanması
    For i = 1 To kps
      If i = 1 Then
        A(3, i) = 0
      ElseIf i >= 2 And i <= 5 Then
        A(3, i) = A(3, i - 1) + (HD(i - 1) - FIP(i - 1) / pt(i -
        1)) * pl(i - 1)
        B(3, i - 1) = A(3, i)
        If i = 5 Then
          B(3, i) = A(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
        End If
      ElseIf i >= 6 And i <= 16 Then
        A(3, i) = B(3, i - 1)
        B(3, i) = A(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
      ElseIf i = 17 Then
        B(3, i) = B(3, 5)
        A(3, i) = B(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
      ElseIf i = 18 Or i = 19 Then
        B(3, i) = A(3, i - 1)
        A(3, i) = B(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
      ElseIf i > 19 Then
        'Kesitin Simetrisi
        A(3, i) = -A(3, i - 19)
        B(3, i) = -B(3, i - 19)
      End If
    Next
  End Sub

```

```

'Kayma Gerilmelerinin Hesaplanacağı Noktaların FIBT değerlerinin
belirlenmesi
For i = 1 To ktn
  If i >= 1 And i <= 5 Then
    FIBT(i) = FIP(i) / pt(i)
  ElseIf i >= 6 And i <= 12 Then
    FIBT(i) = FIP(i - 1) / pt(i - 1)
  ElseIf i >= 13 And i <= 18 Then
    FIBT(i) = FIP(i - 2) / pt(i - 2)
  ElseIf i >= 19 And i <= 22 Then
    FIBT(i) = FIP(i - 3) / pt(i - 3)
  ElseIf i >= 23 And i <= 34 Then
    FIBT(i) = FIP(i - 4) / pt(i - 4)
  ElseIf i >= 35 And i <= 43 Then
    FIBT(i) = FIP(i - 5) / pt(i - 5)
  ElseIf i = 44 Then
    FIBT(i) = FIP(i - 6) / pt(i - 6)
  End If
Next
ElseIf ktip = 5 Then
  '-----
  '*** ÇİFT GÖZLÜ KAPALI KESİT ***
  '-----
  For i = 1 To kps
    Select Case i
      Case 1 To 16:
        FIP(i) = RI(1)
      Case 17 To 22:
        FIP(i) = RI(2) - RI(1)
        If Abs(FIP(i)) < 0.000000001 Then FIP(i) = 0
      Case 23 To 25:
        FIP(i) = 0
      Case 26 To 41:
        FIP(i) = RI(2)
      Case 42 To 44:
        FIP(i) = 0
    End Select
  Next
  'Kesit Parçalarının Dönme Merkezine Dik Mesafeleri
  For i = 1 To kps
    HD(i) = Abs(((A(1, i) - B(1, i)) * A(2, i) - (A(2, i) - B(2,
    i)) * A(1, i)) / (Sqr((A(1, i) - B(1, i)) ^ 2 + (A(2, i) - B(2,
    i)) ^ 2)))
  Next
  'w koordinatlarının hesaplanması
  For i = 1 To kps
    If i = 1 Then
      A(3, i) = 0
    ElseIf i >= 2 And i <= 5 Then
      A(3, i) = A(3, i - 1) + (HD(i - 1) - FIP(i - 1) / pt(i -
      1)) * pl(i - 1)
      B(3, i - 1) = A(3, i)
      If i = 5 Then
        B(3, i) = A(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
      End If
    ElseIf i >= 6 And i <= 16 Then
      A(3, i) = B(3, i - 1)
      B(3, i) = A(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
    ElseIf i >= 17 And i <= 22 Then
      A(3, i) = B(3, i - 1)
      B(3, i) = A(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
    End If
  Next

```

```

ElseIf i = 23 Then
    B(3, i) = B(3, 5)
    A(3, i) = B(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
ElseIf i = 24 Or i = 25 Then
    B(3, i) = A(3, i - 1)
    A(3, i) = B(3, i) + (HD(i) - FIP(i) / pt(i)) * pl(i)
ElseIf i > 25 And i < 42 Then
    'Kesitin Simetrisi
    A(3, i) = -A(3, i - 25)
    B(3, i) = -B(3, i - 25)
ElseIf i >= 42 Then
    A(3, i) = -A(3, i - 19)
    B(3, i) = -B(3, i - 19)
End If
Next
'Kayma Gerilmelerinin Hesaplanacağı Noktaların FIBT değerlerinin
belirlenmesi
For i = 1 To ktn
    If i >= 1 And i <= 5 Then
        FIBT(i) = FIP(i) / pt(i)
    ElseIf i >= 6 And i <= 12 Then
        FIBT(i) = FIP(i - 1) / pt(i - 1)
    ElseIf i >= 13 And i <= 18 Then
        FIBT(i) = FIP(i - 2) / pt(i - 2)
    ElseIf i >= 19 And i <= 25 Then
        FIBT(i) = FIP(i - 3) / pt(i - 3)
    ElseIf i >= 26 And i <= 29 Then
        FIBT(i) = FIP(i - 4) / pt(i - 4)
    ElseIf i >= 30 And i <= 35 Then
        FIBT(i) = FIP(i - 5) / pt(i - 5)
    ElseIf i >= 36 And i <= 42 Then
        FIBT(i) = FIP(i - 6) / pt(i - 6)
    ElseIf i >= 43 And i <= 48 Then
        FIBT(i) = FIP(i - 7) / pt(i - 7)
    ElseIf i >= 49 And i <= 52 Then
        FIBT(i) = FIP(i - 8) / pt(i - 8)
    ElseIf i = 53 Then
        FIBT(i) = FIP(i - 9) / pt(i - 9)
    End If
Next
End If 'Kesit Tipi Tek Gözlü (OR) Çift Gözlü Kapalı Kesit
End Sub

```

```
Public Sub asal_ksen()
```

```

'1.adım : eksenin ağırlık merkezine taşınması
'Kesit Statik momentlerinin bulunması
FX = 0
FY = 0
FW = 0
For i = 1 To kps
    FX = FX + 1 / 2 * pl(i) * pt(i) * (A(1, i) + B(1, i))
    FY = FY + 1 / 2 * pl(i) * pt(i) * (A(2, i) + B(2, i))
    FW = FW + 1 / 2 * pl(i) * pt(i) * (A(3, i) + B(3, i))
Next
'Ağırlık merkezine kaydırılan eksenin ilk eksene olan koordinatları
XO = FX / F
YO = FY / F
WO = FW / F

```

```

If Abs(XO) < 0.000001 Then XO = 0
If Abs(YO) < 0.000001 Then YO = 0
If Abs(WO) < 0.000001 Then WO = 0
'Ağırlık merkezine taşınmış eksene göre
'başlangıç ve bitiş noktalarının yeni koordinatlarının
'hesaplanması
FXX = 0
FYY = 0
FGY = 0
For i = 1 To kps
  A(1, i) = A(1, i) - XO
  A(2, i) = A(2, i) - YO
  A(3, i) = A(3, i) - WO
  B(1, i) = B(1, i) - XO
  B(2, i) = B(2, i) - YO
  B(3, i) = B(3, i) - WO
  'Sektör atalet momentlerinin bulunması
  FXX = FXX + pl(i) * pt(i) / 3 * (A(1, i) ^ 2 + B(1, i) ^ 2 + A(1, i) *
    B(1, i))
  FYY = FYY + pl(i) * pt(i) / 3 * (A(2, i) ^ 2 + B(2, i) ^ 2 + A(2, i) *
    B(2, i))
  'Çarpım atalet momenti
  FGY = FGY + pl(i) * pt(i) / 6 * (2 * A(1, i) * A(2, i) + 2 * B(1, i) *
    B(2, i) + A(1, i) * B(2, i) + A(2, i) * B(1, i))
Next
'Eğer sıfıra çok yakın ise sıfır kabul et.
If Abs(FX) < 0.000001 Then FX = 0
If Abs(FY) < 0.000001 Then FY = 0
If Abs(FGY) < 0.000001 Then FGY = 0

'2.adım: Asal eksen açısının bulunması
If FGY <> 0 Then
  tan2fi = -2 * FGY / (FYY - FXX)
  fi = Atn(tan2fi) / 2
  If fi < 0 Then fi = fi + 1.570796327
  SC(1, 1) = Cos(fi)
  SC(1, 2) = Sin(fi)
  SC(2, 1) = -SC(1, 2)
  SC(2, 2) = SC(1, 1)
  'Asal atalet momentlerinin hesabı
  YR1 = (FXX + FYY) / 2
  YR2 = 1 / 2 * Sqr((FYY - FXX) ^ 2 + 4 * FGY ^ 2)
  FXX = YR1 - YR2
  FYY = YR1 + YR2
  'Başlangıç ve bitiş noktalarının yeni koordinatlarının hesaplanması
  For l = 1 To 2
    For i = 1 To kps
      For n = 1 To 2
        C(l, i) = C(l, i) + SC(l, n) * A(n, i)
        D(l, i) = D(l, i) + SC(l, n) * B(n, i)
      Next
      A(l, i) = C(l, i)
      B(l, i) = D(l, i)
    Next
  Next
End If

```

```

'çarpılma çarpım momentinin hesaplanması
FWX = 0
FWY = 0
For i = 1 To kps
  FWX = FWX + pl(i) * pt(i) / 6 * (A(3, i) * (2 * A(1, i) + B(1, i)) +
    B(3, i) * (A(1, i) + 2 * B(1, i)))
  FWY = FWY + pl(i) * pt(i) / 6 * (A(3, i) * (2 * A(2, i) + B(2, i)) +
    B(3, i) * (A(2, i) + 2 * B(2, i)))
Next
If Abs(FWX) < 0.000001 Then FWX = 0
If Abs(FWY) < 0.000001 Then FWY = 0
' Kayma Merkezinin x ve y koordinatları
'Burada Dx ve Dy Kayma Merkezinin ilk dönme noktası olarak
'Seçilen (A Noktası ) noktaya göre koordinatları
Dx = FWY / FYY
Dy = -FWX / FXX

If FXY <> 0 Then
  Dxo = Dx * SC(1, 1) - Dy * SC(1, 2)
  Dyo = Dx * SC(1, 2) + Dy * SC(1, 1)
End If
XD = Dx + A(1, 1)
YD = Dy + A(2, 1)

FWW = 0
Jb = 0
'Kayma merkezine göre A ve B kesit noktalarının birim çarpılma
koordinatları
For i = 1 To kps
  A(3, i) = A(3, i) + Dy * A(1, i) - Dx * A(2, i)
  B(3, i) = B(3, i) + Dy * B(1, i) - Dx * B(2, i)
  FWW = FWW + pl(i) * pt(i) / 3 * (A(3, i) ^ 2 + B(3, i) ^ 2 + A(3, i) *
    B(3, i))
  Jb = Jb + 1 / 3 * pl(i) * pt(i) ^ 3
Next

FHH = 0
If RO <> 1 Then 'KAPALI KESİT İSE
  For i = 1 To kps
    HD(i) = Abs(((A(2, i) - B(2, i)) * (XD - A(1, i)) - (A(1, i) - B(1,
      i)) * (YD - A(2, i))) / (Sqr((A(2, i) - B(2, i)) ^ 2 + (A(1, i) -
      B(1, i)) ^ 2)))
    FHH = FHH + pl(i) * pt(i) * HD(i) ^ 2
  Next
  If ktip = 4 Then 'Tek Gözlü Kapalı Kesit ise
    ReDim R(1 To 1)
    RI(1) = 2 * Fg(1) / DSBTT
  End If
  For i = 1 To IG
    Jb = Jb + 2 * Fg(i) * RI(i)
  Next
  RO = FHH / (FHH - Jb)
End If
If G = 0 Then G = 8000000 'Birimi: t/m2
If E = 0 Then E = 21000000 'Birimi: t/m2
CK = Sqr(G * Jb / (RO * E * FWW))

```

'Kesit noktaları X,Y,W koordinatlarının bulunması

```

If ktip = 1 Then
  '*** I Kesit ***
  For i = 1 To kns
    If i <= 5 Then
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    Else
      X(i) = B(1, i - 1)
      Y(i) = B(2, i - 1)
      W(i) = B(3, i - 1)
    End If
  Next
ElseIf ktip = 2 Then
  '*** C Kesit ***
  For i = 1 To kns
    If i = 9 Or i = 17 Then
      X(i) = B(1, i - 1)
      Y(i) = B(2, i - 1)
      W(i) = B(3, i - 1)
    Else
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    End If
  Next
ElseIf ktip = 3 Then
  '*** Z Kesit ***
  For i = 1 To kns
    If i = 9 Or i = 17 Then
      X(i) = B(1, i - 1)
      Y(i) = B(2, i - 1)
      W(i) = B(3, i - 1)
    Else
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    End If
  Next
ElseIf ktip = 4 Then
  '*** Tek Gözlü Kapalı Kesit ***
  For i = 1 To kns
    If i >= 1 And i <= 16 Then
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    ElseIf i = 17 Then
      X(i) = B(1, i - 1)
      Y(i) = B(2, i - 1)
      W(i) = B(3, i - 1)
    ElseIf i >= 18 And i <= 20 Then
      X(i) = A(1, i - 1)
      Y(i) = A(2, i - 1)
      W(i) = A(3, i - 1)
    ElseIf i >= 21 Then
      X(i) = A(1, i)
      Y(i) = A(2, i)
      W(i) = A(3, i)
    End If
  Next
ElseIf ktip = 5 Then

```

```

*** Çift Gözlü Kapalı Kesit ***
For i = 1 To kns
  If i >= 1 And i <= 25 Then
    X(i) = A(1, i)
    Y(i) = A(2, i)
    W(i) = A(3, i)
  ElseIf i >= 26 And i <= 40 Then
    X(i) = B(1, i)
    Y(i) = B(2, i)
    W(i) = B(3, i)
  ElseIf i >= 41 And i <= 43 Then
    X(i) = A(1, i + 1)
    Y(i) = A(2, i + 1)
    W(i) = A(3, i + 1)
  End If
Next
End If
End Sub

```

```

Public Sub VX_MATRIS()
'*****
'* BAŞLANGIÇ DEĞERLERİ MATRİSİ *
'*****

Redim VX(3, 1)

Select Case mesnet(0)
  Case 1: 'çatal mesnet sınır şartları
    For i = 0 To 1
      For j = 0 To 3
        VX(j, i) = 0
        If i = 0 And j = 1 Then VX(j, i) = 1           'Çatal Mesnet
        If i = 1 And j = 3 Then VX(j, i) = 1           'fi=0 ve B=0
      Next
    Next

  Case 2: 'sertbest uç sınır şartları
    For i = 0 To 1
      For j = 0 To 3
        VX(j, i) = 0
        If i = 1 And j = 0 Then VX(j, i) = 1           'Sertbest uç -
        B=0 ve MB=0
        If i = 0 And j = 1 Then VX(j, i) = 1
      Next
    Next

  Case 3: 'Ankastre mesnet sınır şartları
    For i = 0 To 1
      For j = 0 To 3
        VX(j, i) = 0
        If i = 0 And j = 2 Then VX(j, i) = 1           'Ankastre
        Mesnet - fi ve fi' = 0
        If i = 1 And j = 3 Then VX(j, i) = 1
      Next
    Next
End Select
End Sub

```

```

Public Sub SISTEM()
'*****
'* GEÇİŞ MATRİSİNİN OLUŞTURULMASI *
'*****

ReDim SVB(aciklik - 1, 4, 3)
ReDim SH(aciklik - 1)
ReDim CH(aciklik - 1)
ReDim SHEK(aciklik - 1)
ReDim CHEK(aciklik - 1)
ReDim SVBEK(aciklik - 1, 4, 3)

For i = 0 To aciklik - 1
  'açıklıklar için sinh ve cosh fonksiyonları
  SH(i) = (Exp(CK * CL(i)) - Exp((-1) * CK * CL(i))) / 2
  CH(i) = (Exp(CK * CL(i)) + Exp((-1) * CK * CL(i))) / 2

  'Geçiş matrisinin
  '1. Satırı
  SVB(i, 0, 0) = 1
  SVB(i, 0, 1) = SH(i) / (RO * CK)
  SVB(i, 0, 2) = 1 - CH(i)
  SVB(i, 0, 3) = CL(i) - SH(i) / (RO * CK)
  '2.Satırı
  SVB(i, 1, 0) = 0
  SVB(i, 1, 1) = CH(i) / RO
  SVB(i, 1, 2) = -CK * SH(i)
  SVB(i, 1, 3) = 1 - CH(i) / RO
  '3.Satırı
  SVB(i, 2, 0) = 0
  SVB(i, 2, 1) = -SH(i) / (RO * CK)
  SVB(i, 2, 2) = CH(i)
  SVB(i, 2, 3) = SH(i) / (RO * CK)
  '4.Satırı
  SVB(i, 3, 0) = 0
  SVB(i, 3, 1) = 0
  SVB(i, 3, 2) = 0
  SVB(i, 3, 3) = 1
  If RO <> 1 Then
    '5.Satırı (Kapalı Kesit ise)
    SVB(i, 4, 0) = 0
    SVB(i, 4, 1) = CH(i)
    SVB(i, 4, 2) = -RO * CK * SH(i)
    SVB(i, 4, 3) = 1 - CH(i)
  Else
    SVB(i, 4, 0) = 0
    SVB(i, 4, 1) = 0
    SVB(i, 4, 2) = 0
    SVB(i, 4, 3) = 0
  End If
  If Ytip(i) = 3 Then
    'Yayıllı Burulma momenti yüklemesinde, yüklemenden ( sonraki kısım
    için )
    'açıklıklar için sinh ve cosh fonksiyonları
    SHEK(i) = (Exp(CK * (CL(i) - ZETA(i))) - Exp((-1) * CK * (CL(i) -
    ZETA(i)))) / 2
    CHEK(i) = (Exp(CK * (CL(i) - ZETA(i))) + Exp((-1) * CK * (CL(i) -
    ZETA(i)))) / 2
    'Geçiş matrisinin
    '1. Satırı
    SVBEK(i, 0, 0) = 1
    SVBEK(i, 0, 1) = SHEK(i) / (RO * CK)
    SVBEK(i, 0, 2) = 1 - CHEK(i)
  End If
End For
End Sub

```

```

SVBEK(i, 0, 3) = (CL(i) - ZETA(i)) - SHEK(i) / (RO * CK)
'2.Satır1
SVBEK(i, 1, 0) = 0
SVBEK(i, 1, 1) = CHEK(i) / RO
SVBEK(i, 1, 2) = -CK * SHEK(i)
SVBEK(i, 1, 3) = 1 - CHEK(i) / RO
'3.Satır1
SVBEK(i, 2, 0) = 0
SVBEK(i, 2, 1) = -SHEK(i) / (RO * CK)
SVBEK(i, 2, 2) = CHEK(i)
SVBEK(i, 2, 3) = SHEK(i) / (RO * CK)
'4.Satır1
SVBEK(i, 3, 0) = 0
SVBEK(i, 3, 1) = 0
SVBEK(i, 3, 2) = 0
SVBEK(i, 3, 3) = 1
  If RO <> 1 Then          '5.Satır1 (Kapalı Kesit ise)
    SVBEK(i, 4, 0) = 0
    SVBEK(i, 4, 1) = CHEK(i)
    SVBEK(i, 4, 2) = -RO * CK * SHEK(i)
    SVBEK(i, 4, 3) = 1 - CHEK(i)
  Else
    SVBEK(i, 4, 0) = 0
    SVBEK(i, 4, 1) = 0
    SVBEK(i, 4, 2) = 0
    SVBEK(i, 4, 3) = 0
  End If
End If
Next 'Açıklık
End Sub

Public Sub VP_yuk_matrisi()
'*****
'* YÜK MATRİSİNİN OLUŞTURULMASI *
'*****
  ReDim VP(aciklik - 1, 4)
  ReDim VPEK(aciklik - 1, 4)
  ReDim SHA(aciklik - 1)
  ReDim CHA(aciklik - 1)
  ReDim SHB(aciklik - 1)
  ReDim CHB(aciklik - 1)
  ReDim SHC(aciklik - 1)
  ReDim CHC(aciklik - 1)

  For i = 0 To aciklik - 1
    'Yükün Bulunduğu Noktadan Mesnete kadar olan kısım için sinh ve cosh
    fonksiyonları
    SHA(i) = (Exp(CK * (CL(i) - ZETA(i))) - Exp(-CK * (CL(i) - ZETA(i))))
    / 2
    CHA(i) = (Exp(CK * (CL(i) - ZETA(i))) + Exp(-CK * (CL(i) - ZETA(i))))
    / 2
    'Farklı yükleme tipleri için yük matrisleri
    If Ytip(i) = 1 Then      'Münferit Burulma Momenti Yükleme
      VP(i, 0) = -YUK(i) * (CL(i) - ZETA(i) - SHA(i) / (RO * CK))
      VP(i, 1) = -YUK(i) * (1 - CHA(i) / RO)
      VP(i, 2) = -YUK(i) * SHA(i) / (RO * CK)
      VP(i, 3) = -YUK(i)
      VP(i, 4) = 0
      If RO <> 1 Then
        VP(i, 4) = -YUK(i) * (1 - CHA(i))
      End If
    End If
  End For
End Sub

```

```

ElseIf Ytip(i) = 2 Then 'Münferit Bimoment Yüklemesi
  VP(i, 0) = -YUK(i) * (1 - CHA(i))
  VP(i, 1) = YUK(i) * (CK * SHA(i))
  VP(i, 2) = -YUK(i) * CHA(i)
  VP(i, 3) = 0
  VP(i, 4) = 0
  If RO <> 1 Then
    VP(i, 4) = YUK(i) * RO * CK * SHA(i)
  End If
ElseIf Ytip(i) = 3 Then 'Yayıllı Burulma Momenti Yüklemesi
  'Yayıllı Burulma Momenti İçin sinh ve cosh fonksiyonları
  SHB(i) = (Exp(CK * (ZETA(i) - CZ(i))) - Exp((-1) * CK * (ZETA(i) -
    CZ(i)))) / 2
  CHB(i) = (Exp(CK * (ZETA(i) - CZ(i))) + Exp((-1) * CK * (ZETA(i) -
    CZ(i)))) / 2
  'Yük Matrisinin elde edilmesi (Yük Sınırları içinde kalması hali)
  VPEK(i, 0) = -YUK(i) * (((ZETA(i) - CZ(i)) ^ 2) / 2 + (1 - CHB(i))
    / (RO * CK ^ 2))
  VPEK(i, 1) = -YUK(i) * (ZETA(i) - CZ(i) - SHB(i) / (RO * CK))
  VPEK(i, 2) = YUK(i) / (RO * CK ^ 2) * (1 - CHB(i))
  VPEK(i, 3) = -YUK(i) * (ZETA(i) - CZ(i))
  VPEK(i, 4) = 0
  If RO <> 1 Then
    VPEK(i, 4) = -YUK(i) * (ZETA(i) - CZ(i) - SHB(i) / CK)
  End If
  'Yayıllı burulma momentinin, bulunduğu açıklık sonuna taşınması
  'Açıklık sonunda yük vektörünün elde edilmesi
  For m = 0 To 4
    For n = 0 To 3
      VP(i, m) = VP(i, m) + SVBEK(i, m, n) * VPEK(i, n)
    Next
  Next
End If
Next
End Sub

Public Sub K_mesnet_matrisi()
  '*****
  'SÜREKSİZLİK MATRİSİNİN OLUŞTURULMASI
  '*****
  ReDim K(aciklik - 2, 4)
  For i = 0 To aciklik - 2
    For j = 0 To 4
      K(i, j) = 0
      If mesnet(i + 1) = 1 Then K(i, 3) = 1 'Çatal Mesnet
    Next
  Next
End Sub

```

```

Public Sub DENKLEMLER()
  ReDim SK(aciklik - 2, aciklik - 2, 4)
  ReDim SVP(aciklik - 1, aciklik - 1, 4)
  ReDim SVX(aciklik - 1, 4, 1)

  'BAŞLANGIÇ DEĞERLERİNİN TAŞINMASI
  '1. Açıklık için Taşıma İşlemi
  For l = 0 To 1
    For m = 0 To 4
      For n = 0 To 3
         $SVX(0, m, l) = SVX(0, m, l) + SVB(0, m, n) * VX(n, l)$ 
      Next
    Next
  Next
  'Diğer Açıklıklar için Taşıma İşlemi
  For i = 1 To aciklik - 1
    For l = 0 To 1
      For m = 0 To 4
        For n = 0 To 3
           $SVX(i, m, l) = SVX(i, m, l) + SVB(i, m, n) * SVX(i - 1, n, l)$ 
        Next 'n
      Next 'm
    Next 'l
  Next 'i

  'SÜREKSİZLİK VEKTÖRLERİNİN TAŞINMASI
  'İlk Ara Mesnetten Sonra Taşınması

  For j = 1 To aciklik - 1      'Herbir Ara mesnet
    For n = 0 To 4
      For m = 0 To 3
         $SK(0, j - 1, n) = SK(0, j - 1, n) + SVB(j, n, m) * K(j - 1, m)$ 
      Next
    Next
  Next
  'Süreksizlik Matrislerinin Sistem Sonuna Kadar Taşınması
  For i = 1 To aciklik - 2
    j = 0
    l = i + 1
    Do While j <= (aciklik - 2 - i)
      For n = 0 To 4
        For m = 0 To 3
           $SK(i, j, n) = SK(i, j, n) + SVB(l, n, m) * SK(i - 1, j, m)$ 
        Next
      Next
      j = j + 1
      l = l + 1
    Loop
  Next

  'YÜK VEKTÖRLERİNİN TAŞINMASI
  'İlk Yük Vektörünün Taşınması
  For i = 0 To aciklik - 1
    For n = 0 To 4
       $SVP(0, i, n) = VP(i, n)$ 
    Next
  Next
Next

```

```

'Yük Vektörlerinin Sistem Sonuna Kadar Taşınması
For i = 1 To aciklik - 1
  j = 0
  l = i
  Do While j + 1 <= aciklik - i
    For n = 0 To 4
      For m = 0 To 3
        SVP(i, j, n) = SVP(i, j, n) + SVB(l, n, m) * SVP(i - 1, j, m)
      Next
    Next
    j = j + 1
    l = l + 1
    If i = aciklik - 1 And j = aciklik - 2 Then j = j + 1
  Loop
Next

```

End Sub

```

Public Sub sistem_cozum()
  ReDim KM(aciklik, aciklik)
  ReDim KP(aciklik)

  'LİNEER DENKLEM TAKIMI ÇÖZÜMÜ İÇİN KATSAYILAR MATRİSİNİN OLUŞTURULMASI
  'HERBİR ARA MESNET NOKTASINDA BİR DENKLEM
  'SON MESNETTE İKİ DENKLEM ELDE EDİLİR

```

```

'Başlangıç Değerlerinin KM matrisine aktarılması
If aciklik > 1 Then
  For i = 0 To aciklik - 2
    If mesnet(i + 1) = 1 Then      'Çatal Mesnet
      For n = 0 To 1
        KM(i, n) = SVX(i, 0, n)
      Next
    ElseIf mesnet(i + 1) = 3 Then  'Ankastre mesnet
      For n = 0 To 1
        KM(i, n) = SVX(i, 0, n)
      Next
    End If
  Next
End If
Next
End If
'Son Mesnet için 2 Denklem Elde Edilir
For n = 0 To 1
  If mesnet(aciklik) = 1 Then      'Çatal Mesnet
    KM(aciklik - 1, n) = SVX(aciklik - 1, 0, n)
    KM(aciklik, n) = SVX(aciklik - 1, 2, n)
  ElseIf mesnet(aciklik) = 2 Then  'Sertbest Uç
    KM(aciklik - 1, n) = SVX(aciklik - 1, 2, n)
    KM(aciklik, n) = SVX(aciklik - 1, 3, n)
  ElseIf mesnet(aciklik) = 3 Then  'Ankastre Mesnet
    KM(aciklik - 1, n) = SVX(aciklik - 1, 0, n)
    KM(aciklik, n) = SVX(aciklik - 1, 1, n)
  End If
Next

```

```

'Ara Mesnet Süreksizliklerin KM Matrisine Eklenmesi
For i = 2 To aciklik
  j = i - 1
  m = 0

```

```

Do While j <= aciklik - 1
  If j < aciklik - 1 Then
    KM(j, i) = SK(m, i - 2, 0)
  Else
    'Mesnet Şartlarına Göre Ayrılacak
    If mesnet(aciklik) = 1 Then      'Çatal Mesnet
      KM(j, i) = SK(m, i - 2, 0)
      KM(j + 1, i) = SK(m, i - 2, 2)
    ElseIf mesnet(aciklik) = 2 Then 'Serbest Uç
      KM(j, i) = SK(m, i - 2, 2)
      KM(j + 1, i) = SK(m, i - 2, 3)
    ElseIf mesnet(aciklik) = 3 Then 'Ankastre Mesnet
      KM(j, i) = SK(m, i - 2, 0)
      KM(j + 1, i) = SK(m, i - 2, 1)
    End If
  End If
  j = j + 1
  m = m + 1
Loop
Next

```

'Yük Vektörlerinin Denklem Takımına Eklenmesi

```

For i = aciklik - 1 To 0 Step -1
  m = i
  n = 0
  Do While n <= i
    If i = aciklik - 1 Then      'Son Açıklıkta ise
      If mesnet(aciklik) = 1 Then 'Çatal Mesnet
        KP(i) = KP(i) + (-1) * SVP(m, n, 0)
        KP(i + 1) = KP(i + 1) + (-1) * SVP(m, n, 2)
      ElseIf mesnet(aciklik) = 2 Then 'Serbest Uç
        KP(i) = KP(i) + (-1) * SVP(m, n, 2)
        KP(i + 1) = KP(i + 1) + (-1) * SVP(m, n, 3)
      ElseIf mesnet(aciklik) = 3 Then 'Ankastre Mesnet
        KP(i) = KP(i) + (-1) * SVP(m, n, 0)
        KP(i + 1) = KP(i + 1) + (-1) * SVP(m, n, 1)
      End If
    Else      'Son Açıklıkta Değilse
      KP(i) = KP(i) + (-1) * SVP(m, n, 0)
    End If
    m = m - 1
    n = n + 1
  Loop
Next

```

End Sub

Public Sub Matris\_Inverse()

'Uses Gauss elimination method in order to calculate the inverse matrix [A]<sup>-1</sup>

'Method: Puts matrix [A] at the left and the singular matrix [I] at the right:

'[ a11 a12 a13 | 1 0 0 ]

'[ a21 a22 a23 | 0 1 0 ]

'[ a31 a32 a33 | 0 0 1 ]

'Then using line operations, we try to build the singular matrix [I] at the left.

'After we have finished, the inverse matrix [A]<sup>-1</sup> (bij) will be at the right:

```
'[ 1 0 0 | b11 b12 b13 ]
'[ 0 1 0 | b21 b22 b23 ]
'[ 0 0 1 | b31 b32 b33 ]
```

```
Dim System_DIM As Integer 'Current Matrix [A] dimensions
Dim Matrix_A()
Dim Operations_Matrix() 'Matrix where the calculations are done
Dim Inverse_Matrix() 'Matrix with the Inverse of [A]
Dim MAX_DIM As Double
```

```
ReDim Matrix_A(1 To aciklik + 1, 1 To aciklik + 1)
ReDim Operations_Matrix(1 To aciklik + 1, 1 To (aciklik + 1) * 2) 'Matrix
    where the calculations are done
ReDim Inverse_Matrix(1 To aciklik + 1, 1 To aciklik + 1) 'Matrix with the
    Inverse of [A]
```

```
Solution_Problem = False
```

```
On Error GoTo errhandler 'In case the inverse cannot be found
    (Determinant = 0)
```

```
System_DIM = aciklik + 1
MAX_DIM = aciklik + 1
```

```
For i = 0 To aciklik
    For j = 0 To aciklik
        Matrix_A(i + 1, j + 1) = KM(i, j)
    Next
Next
```

```
'Assign values from matrix [A] at the left
For nu = 1 To System_DIM
    For mu = 1 To System_DIM
        Operations_Matrix(mu, nu) = Matrix_A(mu, nu)
    Next
Next
```

```
'Assign values from singular matrix [I] at the right
For nu = 1 To System_DIM
    For mu = 1 To System_DIM
        If nu = mu Then
            Operations_Matrix(mu, nu + System_DIM) = 1
        Else
            Operations_Matrix(mu, nu + System_DIM) = 0
        End If
    Next
Next
```

```
'Build the Singular matrix [I] at the left
For ku = 1 To System_DIM
    'Bring a non-zero element first by changes lines if necessary
    If Operations_Matrix(ku, ku) = 0 Then
        For nu = ku To System_DIM
            If Operations_Matrix(nu, ku) <> 0 Then line_1 = nu: Exit For
        Next nu
        'Finds line_1 with non-zero element
        'Change line k with line_1
        For mu = ku To System_DIM * 2
            temporary_1 = Operations_Matrix(ku, mu)
            Operations_Matrix(ku, mu) = Operations_Matrix(line_1, mu)
            Operations_Matrix(line_1, mu) = temporary_1
        Next mu
    End If
```

```

    elem1 = Operations_Matrix(ku, ku)
    For nu = ku To 2 * System_DIM
        Operations_Matrix(ku, nu) = Operations_Matrix(ku, nu) / elem1
    Next nu

```

```

'For other lines, make a zero element by using:
'Ai1=Aij-A11*(Aij/A11)
'and change all the line using the same formula for other elements
For nu = 1 To System_DIM
    If nu = ku And nu = MAX_DIM Then Exit For 'Finished
    If nu = ku And nu < MAX_DIM Then nu = nu + 1 'Do not change
    that element (already equals to 1), go for next one
    If Operations_Matrix(nu, ku) <> 0 Then 'if it is zero, stays as
    it is
        multiplier_1 = Operations_Matrix(nu, ku) /
        Operations_Matrix(ku, ku)
        For mu = ku To 2 * System_DIM
            Operations_Matrix(nu, mu) = Operations_Matrix(nu, mu) -
            Operations_Matrix(ku, mu) * multiplier_1
        Next mu
    End If
Next nu
Next ku

```

```

'Assign the right part to the Inverse_Matrix
For nu = 1 To System_DIM
    For ku = 1 To System_DIM
        Inverse_Matrix(nu, ku) = Operations_Matrix(nu, System_DIM + ku)
    Next ku
Next nu

```

```

ReDim KMTERS(aciklik, aciklik)
'Ters Matrisin KMTERS matrisine aktarılması
For i = 0 To aciklik
    For j = 0 To aciklik
        KMTERS(i, j) = Inverse_Matrix(i + 1, j + 1)
    Next
Next
Exit Sub

```

```

errhandler:
message$ = "Katsayılar Matrisinin Tersı Alınırken Bir Hata Oluştı!" &
    Chr(13) & "Katsayılar Matrisinin Determinantı Sıfır Olabilir!"
response = MsgBox(message$, vbCritical)
Solution_Problem = True

End Sub

```

```

Public Sub bilinmeyen_bul()
  ReDim BILBUL(aciklik)
  ReDim VXX(3)

  For i = 0 To aciklik
    For j = 0 To aciklik
      BILBUL(i) = BILBUL(i) + KMTERS(i, j) * KP(j)
    Next
  Next

  'Başlangıç Değerleri Matrisine Bilinmeyenlerin Yerleştirilmesi
  Select Case mesnet(0)
    Case 1: 'çatal mesnet sınır şartları
      VXX(0) = 0
      VXX(1) = BILBUL(0)
      VXX(2) = 0
      VXX(3) = BILBUL(1)
    Case 2: 'sertbest uç sınır şartları
      VXX(0) = BILBUL(0)
      VXX(1) = BILBUL(1)
      VXX(2) = 0
      VXX(3) = 0
    Case 3: 'Ankastre mesnet sınır şartları
      VXX(0) = 0
      VXX(1) = 0
      VXX(2) = BILBUL(0)
      VXX(3) = BILBUL(1)
  End Select

  'Ara Mesnetlerdeki Bilinmeyenlerin
  'Süreksizlik Matrislerine Yerleştirilmesi

  For i = 0 To aciklik - 2
    For j = 0 To 4
      K(i, j) = 0
    Next
    If mesnet(i + 1) = 1 Then K(i, 3) = BILBUL(i + 2)
    If mesnet(i + 1) = 3 Then K(i, 3) = BILBUL(i + 2)
  Next
End Sub

Public Sub Kesit_Tesirleri()
  '*****
  '*** Kesit Tesirlerinin Bulunması ***
  '*****

  Dim VPP() As Double
  Dim VPPY() As Double
  Dim KPP() As Double
  Dim sayac As Integer
  Dim sycyuk As Integer
  Dim SinhB() As Double
  Dim CoshB() As Double
  Dim SinhC() As Double
  Dim CoshC() As Double

  If CPS = 0 Then CPS = 12
  'münferit yükleme için kesit hesap noktaları
  Pn = CPS + 1
  PPn = (Pn - 1) / 2
  'yayıllı burulma momenti için kesit hesap noktaları
  Pny = CPS + 2
  PPny = Pny - 2

```

```

ReDim KPP(aciklik - 2, 4)
ReDim VPP(aciklik - 1, 4)
ReDim VPPY(aciklik - 1, Pny, 4)
ReDim SNH(aciklik - 1, Pn + 1)
ReDim CSH(aciklik - 1, Pn + 1)
ReDim SNHA(aciklik - 1, (CPS + 2))
ReDim CSHA(aciklik - 1, (CPS + 2))
ReDim MBS(aciklik - 1, Pn)
ReDim Z(aciklik - 1, Pn + 1)

```

```
'Z Mesafeleri
```

```

For i = 0 To aciklik - 1
  If Ytip(i) = 3 Then 'Yayılı yükleme durumu
    For j = 0 To Pny
      If j = 0 Then Z(i, j) = 0
      If CZ(i) <> 0 Then
        If j = 1 Then
          Z(i, j) = CZ(i)
          Z(i, j + 1) = CZ(i)
        End If
        If j > 1 And j < PPny Then Z(i, j + 1) = Z(i, j) + (ZETA(i) - CZ(i)) / (CPS - 2)
        If j = PPny Then Z(i, j + 1) = Z(i, j)
        If j = Pny - 1 Then Z(i, j + 1) = Z(i, j) + CL(i) - ZETA(i)
      ElseIf CZ(i) = 0 Then
        If j = 1 Then Z(i, j) = 0
        If j > 1 And j <= PPny Then Z(i, j) = Z(i, j - 1) + ZETA(i) / (CPS - 1)
        If j = PPny Then Z(i, j + 1) = Z(i, j)
        If j = Pny - 1 Then Z(i, j + 1) = Z(i, j) + CL(i) - ZETA(i)
      End If
      SNH(i, j) = (Exp(CK * Z(i, j)) - Exp((-1) * CK * Z(i, j))) / 2
      CSH(i, j) = (Exp(CK * Z(i, j)) + Exp((-1) * CK * Z(i, j))) / 2
      If j > PPny Then
        SNHA(i, j) = (Exp(CK * (Z(i, j) - ZETA(i))) - Exp((-1) * CK * (Z(i, j) - ZETA(i)))) / 2
        CSHA(i, j) = (Exp(CK * (Z(i, j) - ZETA(i))) + Exp((-1) * CK * (Z(i, j) - ZETA(i)))) / 2
      End If
    Next
  Else 'Münferit yükleme durumu
    For j = 0 To Pn
      If j = 0 Then Z(i, j) = 0
      If j < PPn Then Z(i, j + 1) = Z(i, j) + ZETA(i) / (CPS / 2)
      If j = PPn Then Z(i, j + 1) = Z(i, j)
      If j > PPn And j < Pn - 1 Then Z(i, j + 1) = Z(i, j) + (CL(i) - ZETA(i)) / (CPS / 2)
      If j = Pn - 1 Then Z(i, j + 1) = CL(i)
      SNH(i, j) = (Exp(CK * Z(i, j)) - Exp((-1) * CK * Z(i, j))) / 2
      CSH(i, j) = (Exp(CK * Z(i, j)) + Exp((-1) * CK * Z(i, j))) / 2
      If j > PPn Then
        SNHA(i, j) = (Exp(CK * (Z(i, j) - ZETA(i))) - Exp((-1) * CK * (Z(i, j) - ZETA(i)))) / 2
        CSHA(i, j) = (Exp(CK * (Z(i, j) - ZETA(i))) + Exp((-1) * CK * (Z(i, j) - ZETA(i)))) / 2
      End If
    Next
  End If
Next
Next 'Z mesafeleri - aciklik

```

```

ReDim VB(aciklik - 1, Pny, 4, 3)
ReDim VBA(aciklik - 1, ((Pn + 1) / 2), 4, 3)

'Geçiş Matrisinin Yazılması - Kesit Noktaları için
For i = 0 To aciklik - 1
  If Ytip(i) = 3 Then
    sayac = Pny
    sycyuk = PPny
  Else
    sayac = Pn
    sycyuk = PPn
  End If

  For j = 0 To sayac
    'Geçiş Matrisinin
    '1. Satırını
    VB(i, j, 0, 0) = 1
    VB(i, j, 0, 1) = SNH(i, j) / (RO * CK)
    VB(i, j, 0, 2) = 1 - CSH(i, j)
    VB(i, j, 0, 3) = Z(i, j) - SNH(i, j) / (RO * CK)
    '2.Satırını
    VB(i, j, 1, 0) = 0
    VB(i, j, 1, 1) = CSH(i, j) / RO
    VB(i, j, 1, 2) = -CK * SNH(i, j)
    VB(i, j, 1, 3) = 1 - CSH(i, j) / RO
    '3.Satırını
    VB(i, j, 2, 0) = 0
    VB(i, j, 2, 1) = -SNH(i, j) / (RO * CK)
    VB(i, j, 2, 2) = CSH(i, j)
    VB(i, j, 2, 3) = SNH(i, j) / (RO * CK)
    '4.Satırını
    VB(i, j, 3, 0) = 0
    VB(i, j, 3, 1) = 0
    VB(i, j, 3, 2) = 0
    VB(i, j, 3, 3) = 1
    '5.Satırını (Kapalı Kesit ise)
    If RO <> 1 Then
      VB(i, j, 4, 0) = 0
      VB(i, j, 4, 1) = CSH(i, j)
      VB(i, j, 4, 2) = -RO * CK * SNH(i, j)
      VB(i, j, 4, 3) = 1 - CSH(i, j)
    Else
      VB(i, j, 4, 0) = 0
      VB(i, j, 4, 1) = 0
      VB(i, j, 4, 2) = 0
      VB(i, j, 4, 3) = 0
    End If

    'Yükten Sonraki Kısım
    If j > sycyuk Then
      If Ytip(i) <> 3 Then
        m = (j - (CPS + 2) / 2)
      Else
        m = (j - PPny - 1)
      End If
      '1. Satırını
      VBA(i, m, 0, 0) = 1
      VBA(i, m, 0, 1) = SNHA(i, j) / (RO * CK)
      VBA(i, m, 0, 2) = 1 - CSHA(i, j)
      VBA(i, m, 0, 3) = (Z(i, j) - ZETA(i)) - SNHA(i, j) / (RO * CK)
    End If
  Next j
Next i

```

```

'2.Satırı
VBA(i, m, 1, 0) = 0
VBA(i, m, 1, 1) = CSHA(i, j) / RO
VBA(i, m, 1, 2) = -CK * SNHA(i, j)
VBA(i, m, 1, 3) = 1 - CSHA(i, j) / RO
'3.Satırı
VBA(i, m, 2, 0) = 0
VBA(i, m, 2, 1) = -SNHA(i, j) / (RO * CK)
VBA(i, m, 2, 2) = CSHA(i, j)
VBA(i, m, 2, 3) = SNHA(i, j) / (RO * CK)
'4.Satırı
VBA(i, m, 3, 0) = 0
VBA(i, m, 3, 1) = 0
VBA(i, m, 3, 2) = 0
VBA(i, m, 3, 3) = 1
'5.Satırı (Kapalı Kesit ise)
If RO <> 1 Then
  VBA(i, m, 4, 0) = 0
  VBA(i, m, 4, 1) = CSHA(i, j)
  VBA(i, m, 4, 2) = -RO * CK * SNHA(i, j)
  VBA(i, m, 4, 3) = 1 - CSHA(i, j)
Else
  VBA(i, m, 4, 0) = 0
  VBA(i, m, 4, 1) = 0
  VBA(i, m, 4, 2) = 0
  VBA(i, m, 4, 3) = 0
End If
End If      'Yükten Sonraki Kısım
Next 'j
Next 'i

ReDim SinhB(aciklik - 1, Pny)
ReDim CoshB(aciklik - 1, Pny)
ReDim SinhC(aciklik - 1, Pny)
ReDim CoshC(aciklik - 1, Pny)

'Yüklemelerin Eklenmesi
For i = 0 To aciklik - 1
  If Ytip(i) = 3 Then
    sycyuk = PPny
  Else
    sycyuk = PPn
  End If

  If Ytip(i) = 1 Then      'Münferit Burulma Momenti
    VPP(i, 0) = 0
    VPP(i, 1) = 0
    VPP(i, 2) = 0
    VPP(i, 3) = -YUK(i)
  ElseIf Ytip(i) = 2 Then  'Münferit Bimoment
    VPP(i, 0) = 0
    VPP(i, 1) = 0
    VPP(i, 2) = -YUK(i)
    VPP(i, 3) = 0
  ElseIf Ytip(i) = 3 Then  'Yayıllı Burulma Momenti
    For j = 0 To Pny
      If j = 0 Or j = 1 Then
        VPPY(i, j, 0) = 0
        VPPY(i, j, 1) = 0
        VPPY(i, j, 2) = 0
        VPPY(i, j, 3) = 0
      End If
    Next j
  End If
Next i

```

```

ElseIf j > 1 And j <= PPny Then 'yayıllı yükün etkidiği bölge
  'Yayıllı Burulma Momenti için sinh ve cosh fonksiyonları
  SinhB(i, j) = (Exp(CK * (Z(i, j) - CZ(i))) - Exp((-1) * CK *
  (Z(i, j) - CZ(i)))) / 2
  CoshB(i, j) = (Exp(CK * (Z(i, j) - CZ(i))) + Exp((-1) * CK *
  (Z(i, j) - CZ(i)))) / 2
  'Yük Matrisinin elde edilmesi (Yük Sınırları içinde kalması hali)
  VPPY(i, j, 0) = -YUK(i) * ((Z(i, j) - CZ(i)) ^ 2) / 2 + (1 -
  CoshB(i, j)) / (RO * CK ^ 2)
  VPPY(i, j, 1) = -YUK(i) * (Z(i, j) - CZ(i) - SinhB(i, j) / (RO *
  CK))
  VPPY(i, j, 2) = YUK(i) / (RO * CK ^ 2) * (1 - CoshB(i, j))
  VPPY(i, j, 3) = -YUK(i) * (Z(i, j) - CZ(i))
  VPPY(i, j, 4) = 0
  If RO <> 1 Then
    VPPY(i, j, 4) = -YUK(i) * (Z(i, j) - CZ(i) - SinhB(i, j) / CK)
  End If
  'Yayıllı yükün etkidiği bölgeden sonrası - her nokta için
ElseIf j > PPny Then 'yayıllı yükün etkidiği bölgeden sonra
  SinhB(i, j) = (Exp(CK * (Z(i, j) - CZ(i))) - Exp((-1) * CK *
  (Z(i, j) - CZ(i)))) / 2
  CoshB(i, j) = (Exp(CK * (Z(i, j) - CZ(i))) + Exp((-1) * CK *
  (Z(i, j) - CZ(i)))) / 2
  SinhC(i, j) = (Exp(CK * (Z(i, j) - ZETA(i))) - Exp((-1) * CK *
  (Z(i, j) - ZETA(i)))) / 2
  CoshC(i, j) = (Exp(CK * (Z(i, j) - ZETA(i))) + Exp((-1) * CK *
  (Z(i, j) - ZETA(i)))) / 2
  'Yük Matrisinin elde edilmesi (Yük Sınırları içinde kalması hali)
  VPPY(i, j, 0) = -YUK(i) * ((Z(i, j) - CZ(i)) * (Z(i, j) -
  (ZETA(i) + CZ(i)) / 2) + 1 / (RO * CK ^ 2) * (CoshC(i, j) -
  CoshB(i, j)))
  VPPY(i, j, 1) = -YUK(i) * (ZETA(i) - CZ(i) + 1 / (RO * CK) *
  (SinhC(i, j) - SinhB(i, j)))
  VPPY(i, j, 2) = YUK(i) / (RO * CK ^ 2) * (CoshC(i, j) - CoshB(i,
  j))
  VPPY(i, j, 3) = -YUK(i) * (ZETA(i) - CZ(i))
  VPPY(i, j, 4) = 0
  If RO <> 1 Then
    VPPY(i, j, 4) = -YUK(i) * (ZETA(i) - CZ(i) + 1 / CK * (SinhC(i,
  j) - SinhB(i, j)))
  End If
End If
Next
End If
Next
'Süreksizliklerin Eklenmesi

For i = 0 To aciklik - 2
  KPP(i, 3) = K(i, 3)
Next

```

```

*****
*   V SONUÇ VEKTÖRÜNÜN BULUNMASI   *
*   KESİT TESİRLERİNİN HESABI     *
*****

ReDim V(aciklik - 1, Pny, 4)

For i = 0 To aciklik - 1
  If Ytip(i) <> 3 Then
    *****
    *   TEKİL YÜKLEME   *
    *****
    If i = 0 Then      'İlk Açıklık
      For j = 0 To Pn
        If j <= PPn Then      'Yükten Önceki Kısım
          For n = 0 To 4
            For m = 0 To 3
              V(i, j, n) = V(i, j, n) + VB(i, j, n, m) *
VXX(m)
            Next
          Next
        ElseIf j > PPn Then      'Yükten Sonraki Kısım
          For n = 0 To 4
            For m = 0 To 3
              V(i, j, n) = V(i, j, n) + VB(i, j, n, m) *
VXX(m)
              V(i, j, n) = V(i, j, n) + VBA(i, (j - (CPS
+ 2) / 2), n, m) * VPP(i, m)
            Next m
          Next n
        End If 'Status j
      Next j
    Else      'Diğer Açıklıklar
      For j = 0 To Pn
        If j = 0 Then      'Açıklığın başında bir önceki
açıklığın sonuç vektörünü açıklığın başına aktar
          For m = 0 To 4
            If Ytip(i - 1) = 3 Then
              V(i, 0, m) = V(i, 0, m) + V(i - 1, Pny, m)
            Else
              V(i, 0, m) = V(i, 0, m) + V(i - 1, Pn, m)
            End If
            V(i, 0, m) = V(i, 0, m) + KPP(i - 1, m)
          Next
        Else      'j>0 ise
          If j <= PPn Then      'Yükten Önceki Kısım
            For n = 0 To 4
              For m = 0 To 3
                V(i, j, n) = V(i, j, n) + VB(i, j, n, m) *
V(i, 0, m)
              Next
            Next
          ElseIf j > PPn Then      'Yükten Sonraki Kısım
            For n = 0 To 4

```

```

        For m = 0 To 3
            V(i, j, n) = V(i, j, n) + VB(i, j, n, m) *
V(i, 0, m)
            V(i, j, n) = V(i, j, n) + VBA(i, (j - (CPS +
2) / 2), n, m) * VPP(i, m)
        Next
    Next

    End If

        End If      'j=0 or j>0
    Next j

    End If      'i=0 or diğer açıklıklar

ElseIf Ytip(i) = 3 Then
'*****
'*      YAYILI YÜKLEME      *
'*****

    If i = 0 Then      'ilk açıklık
        If CZ(i) = 0 Then      'cz=0 ise
            For j = 0 To Pny
                If j = 0 Then      'Yayıllı yükün başlangıcından
önceki kısım
                    For n = 0 To 4
                        For m = 0 To 3
                            V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * VXX(m)
                        Next
                    Next
                ElseIf j >= 1 And j <= PPny Then      'Yayıllı yükün
etkidiği bölge
                    For n = 0 To 4
                        For m = 0 To 3
                            V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * VXX(m)
                        Next
                    Next
                    V(i, j, n) = V(i, j, n) + VPPY(i, j, n)
                Next
                ElseIf j > PPny Then      'yayıllı yükten sonraki
bölge
                    For n = 0 To 4
                        For m = 0 To 3
                            V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * VXX(m)
                            'V(i, j, n) = V(i, j, n) + VBA(i, (j -
PPny - 1), n, m) * VPP(i, m)
                        Next
                    Next
                    V(i, j, n) = V(i, j, n) + VPPY(i, j, n)
                Next
            End If
        Next j
    Else      'CZ<>0 ise
        For j = 0 To Pny
            If j <= 1 Then      'yayıllı yükten önceki bölge
                For n = 0 To 4
                    For m = 0 To 3
                        V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * VXX(m)
                    Next
                Next
            End If
        Next
    End If
Next

```

```

ElseIf j > 1 And j <= PPny Then      'yayıllı yükün
etkideği bölge
      For n = 0 To 4
      For m = 0 To 3
          V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * VXX(m)
      Next
      V(i, j, n) = V(i, j, n) + VPPY(i, j, n)
      Next
ElseIf j > PPny Then      'yayıllı yükten sonraki
bölge
      For n = 0 To 4
      For m = 0 To 3
          V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * VXX(m)
          'V(i, j, n) = V(i, j, n) + VBA(i, (j -
PPny - 1), n, m) * VPP(i, m)
      Next
      V(i, j, n) = V(i, j, n) + VPPY(i, j, n)
      Next
      End If
      Next j
      End If 'cz=0 olup olmaması

Else
'*** Diğer açıklıklar ***
  If CZ(i) = 0 Then
    For j = 0 To Pny
      If j = 0 Then      'Açıklığın başında bir önceki
açıklığın sonuç vektörünü açıklığın başına aktar
        For m = 0 To 4
          If Ytip(i - 1) = 3 Then
            V(i, 0, m) = V(i, 0, m) + V(i - 1, Pny,
m)
          Else
            V(i, 0, m) = V(i, 0, m) + V(i - 1, Pn,
m)
          End If
          V(i, 0, m) = V(i, 0, m) + KPP(i - 1, m)
        Next
      ElseIf j >= 1 And j <= PPny Then      'Yayıllı yükün
etkideği bölge
        For n = 0 To 4
        For m = 0 To 3
            V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * V(i, 0, m)
        Next
        V(i, j, n) = V(i, j, n) + VPPY(i, j, n)
        Next
      ElseIf j > PPny Then      'yayıllı yükten sonraki
bölge
        For n = 0 To 4
        For m = 0 To 3
            V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * V(i, 0, m)
        Next
        V(i, j, n) = V(i, j, n) + VPPY(i, j, n)
        Next
      End If      'Yük Bölgesi
    Next j
  Else      'cz<>0 ise
    For j = 0 To Pny

```

```

        If j = 0 Then      'Açıklığın başında bir önceki
açıklığın sonuç vektörünü açıklığın başına aktar
            For m = 0 To 4
                If Ytip(i - 1) = 3 Then
                    V(i, 0, m) = V(i, 0, m) + V(i - 1, Pny,
m)
                Else
                    V(i, 0, m) = V(i, 0, m) + V(i - 1, Pn,
m)
                End If
                V(i, 0, m) = V(i, 0, m) + KPP(i - 1, m)
            Next
        ElseIf j = 1 Then  'yayıllı yükün etkidiği bölgeden
önceki kısım
            For n = 0 To 4
                For m = 0 To 3
                    V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * V(i, 0, m)
                Next
            Next
        ElseIf j > 1 And j <= PPny Then  'Yayıllı yükün
etkidiği bölge
            For n = 0 To 4
                For m = 0 To 3
                    V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * V(i, 0, m)
                Next
                V(i, j, n) = V(i, j, n) + VPPY(i, j, n)
            Next
        ElseIf j > PPny Then  'yayıllı yükten sonraki
bölge
            For n = 0 To 4
                For m = 0 To 3
                    V(i, j, n) = V(i, j, n) + VB(i, j, n,
m) * V(i, 0, m)
                Next
                V(i, j, n) = V(i, j, n) + VPPY(i, j, n)
            Next
        End If 'Yük Bölgesi
    Next j

        End If 'cz=0 veya değil
    End If      'i=0 or diğer açıklıklar
End If      'Yük Tipi
Next i      'aciklik

```

End Sub

```

Public Sub carpilma()
    ReDim TJ(1 To ktn)
    ReDim EK(1 To kps)
    ReDim SWDS(1 To ktn)
    ReDim SWIK(1 To ktn)

    'KESİT PARÇALARININ ÇARPILMA YÜZEYLERİNİN BULUNMASI
    For i = 1 To kps
        EK(i) = pl(i) * pt(i) / 2 * (A(3, i) + B(3, i))
    Next

```

```

If ktip = 1 Then
  '*** I KESİT ***

  'Primer Kayma Gerilmesi Hesaplanacak Noktalarda Kesit Kalınlıkları
  For i = 1 To ktn
    If i >= 1 And i <= 4 Then TJ(i) = pt(i)
    If i >= 5 And i <= 13 Then TJ(i) = pt(i - 1)
    If i >= 14 And i <= 27 Then TJ(i) = pt(i - 3)
    If i = 28 Or i = 29 Then TJ(i) = pt(i - 5)
  Next
  'Sekonder Kayma Gerilmesi Bulunacak Noktaların FW Çarpılma
  Yüzeylerinin Hesaplanması
  SWDS(14) = 0
  SWDS(12) = SWDS(14) - EK(11)
  SWDS(10) = SWDS(12) - EK(9)
  SWDS(8) = SWDS(10) - EK(7)
  SWDS(6) = SWDS(8) - EK(5)
  SWDS(15) = 0
  SWDS(13) = SWDS(15) + EK(12)
  SWDS(11) = SWDS(13) + EK(10)
  SWDS(9) = SWDS(11) + EK(8)
  SWDS(7) = SWDS(9) + EK(6)
  SWDS(5) = SWDS(7) - SWDS(6)
  SWDS(4) = SWDS(5) + EK(4)
  SWDS(3) = SWDS(4) + EK(3)
  SWDS(2) = SWDS(3) + EK(2)
  SWDS(1) = SWDS(2) + EK(1)
  SWDS(28) = 0
  SWDS(26) = SWDS(28) + EK(23)
  SWDS(24) = SWDS(26) + EK(21)
  SWDS(22) = SWDS(24) + EK(19)
  SWDS(20) = SWDS(22) + EK(17)
  SWDS(29) = 0
  SWDS(27) = SWDS(29) - EK(24)
  SWDS(25) = SWDS(27) - EK(22)
  SWDS(23) = SWDS(25) - EK(20)
  SWDS(21) = SWDS(23) - EK(18)
  SWDS(19) = SWDS(20) - SWDS(21)
  SWDS(18) = SWDS(19) - EK(16)
  SWDS(17) = SWDS(18) - EK(15)
  SWDS(16) = SWDS(17) - EK(14)

ElseIf ktip = 2 Then
  '*** C KESİT ***

  'Primer Kayma Gerilmesi Hesaplanacak Noktalarda Kesit Kalınlıkları
  For i = 1 To ktn
    If i >= 1 And i <= 4 Then TJ(i) = pt(i)
    If i >= 5 And i <= 9 Then TJ(i) = pt(i - 1)
    If i >= 10 And i <= 18 Then TJ(i) = pt(i - 2)
    If i = 19 Then TJ(i) = pt(i - 3)
  Next
  'Sekonder Kayma Gerilmesi Bulunacak Noktaların FW Çarpılma
  Yüzeylerinin Hesaplanması
  SWDS(10) = 0
  SWDS(9) = SWDS(10) + EK(8)
  SWDS(8) = SWDS(9) + EK(7)
  SWDS(7) = SWDS(8) + EK(6)
  SWDS(6) = SWDS(7) + EK(5)
  SWDS(5) = SWDS(6)
  SWDS(4) = SWDS(5) - EK(4)
  SWDS(3) = SWDS(4) - EK(3)

```

```

SWDS(2) = SWDS(3) - EK(2)
SWDS(1) = SWDS(2) - EK(1)
SWDS(19) = 0
SWDS(18) = SWDS(19) - EK(16)
SWDS(17) = SWDS(18) - EK(15)
SWDS(16) = SWDS(17) - EK(14)
SWDS(15) = SWDS(16) - EK(13)
SWDS(14) = SWDS(15)
SWDS(13) = SWDS(14) + EK(12)
SWDS(12) = SWDS(13) + EK(11)
SWDS(11) = SWDS(12) + EK(10)
ElseIf ktip = 3 Then
  '*** Z KESİT ***
  'Primer Kayma Gerilmesi Hesaplanacak Noktalarda Kesit Kalınlıkları
  For i = 1 To ktn
    If i >= 1 And i <= 4 Then TJ(i) = pt(i)
    If i >= 5 And i <= 9 Then TJ(i) = pt(i - 1)
    If i >= 10 And i <= 18 Then TJ(i) = pt(i - 2)
    If i = 19 Then TJ(i) = pt(i - 3)
  Next
  'Sekonder Kayma Gerilmesi Bulunacak Noktaların FW Çarpılma
  Yüzeylerinin Hesaplanması
  SWDS(10) = 0
  SWDS(9) = SWDS(10) + EK(8)
  SWDS(8) = SWDS(9) + EK(7)
  SWDS(7) = SWDS(8) + EK(6)
  SWDS(6) = SWDS(7) + EK(5)
  SWDS(5) = SWDS(6)
  SWDS(19) = 0
  SWDS(18) = SWDS(19) + EK(16)
  SWDS(17) = SWDS(18) + EK(15)
  SWDS(16) = SWDS(17) + EK(14)
  SWDS(15) = SWDS(16) + EK(13)
  SWDS(14) = SWDS(15)
  If bust = balt Then
    SWDS(4) = SWDS(5)
    SWDS(3) = SWDS(4)
    SWDS(2) = SWDS(3)
    SWDS(1) = SWDS(2)
    SWDS(13) = SWDS(14)
    SWDS(12) = SWDS(13)
    SWDS(11) = SWDS(12)
  ElseIf bust > balt Then
    SWDS(4) = SWDS(5) + EK(4)
    SWDS(3) = SWDS(4) + EK(3)
    SWDS(2) = SWDS(3) + EK(2)
    SWDS(1) = SWDS(2) + EK(1)
    SWDS(13) = SWDS(14) - EK(12)
    SWDS(12) = SWDS(13) - EK(11)
    SWDS(11) = SWDS(12) - EK(10)
  Else
    SWDS(4) = SWDS(5) - EK(4)
    SWDS(3) = SWDS(4) - EK(3)
    SWDS(2) = SWDS(3) - EK(2)
    SWDS(1) = SWDS(2) - EK(1)
    SWDS(13) = SWDS(14) + EK(12)
    SWDS(12) = SWDS(13) + EK(11)
    SWDS(11) = SWDS(12) + EK(10)
  End If

```

```

ElseIf ktip = 4 Then
  '*** Tek Gözlü Kapalı Kesit ***
  'Primer Kayma Gerilmesi Hesaplanacak Noktalarda Kesit Kalınlıkları
  For i = 1 To ktn
    If i >= 1 And i <= 5 Then TJ(i) = pt(i)
    If i >= 6 And i <= 12 Then TJ(i) = pt(i - 1)
    If i >= 13 And i <= 18 Then TJ(i) = pt(i - 2)
    If i >= 19 And i <= 22 Then TJ(i) = pt(i - 3)
    If i >= 23 And i <= 34 Then TJ(i) = pt(i - 4)
    If i >= 35 And i <= 43 Then TJ(i) = pt(i - 5)
    If i = 44 Then TJ(i) = pt(i - 6)
  Next
  'Sekonder Kayma Gerilmesi Bulunacak Noktaların FW Çarpılma
  'Yüzeylerinin Hesaplanması
  SWDS(1) = 0
  For i = 2 To 6
    SWDS(i) = SWDS(i - 1) + EK(i - 1)
  Next
  SWDS(23) = 0
  SWDS(22) = SWDS(23) - EK(19)
  SWDS(21) = SWDS(22) - EK(18)
  SWDS(20) = SWDS(21) - EK(17)
  SWDS(7) = SWDS(6) - SWDS(20)
  SWDS(8) = SWDS(7) + EK(6)
  SWDS(9) = SWDS(8) + EK(7)
  SWDS(10) = SWDS(9) + EK(8)
  SWDS(11) = SWDS(10) + EK(9)
  SWDS(12) = SWDS(11) + EK(10)
  SWDS(13) = SWDS(12) + EK(11)
  SWDS(14) = SWDS(13)
  For i = 15 To 19
    SWDS(i) = SWDS(i - 1) + EK(i - 3)
  Next
  SWDS(24) = SWDS(1) - EK(20)
  For i = 25 To 28
    SWDS(i) = SWDS(i - 1) - EK(i - 4)
  Next
  SWDS(44) = 0
  SWDS(43) = SWDS(44) + EK(38)
  SWDS(42) = SWDS(43) + EK(37)
  SWDS(41) = SWDS(42) + EK(36)
  SWDS(29) = SWDS(28) - SWDS(41)
  For i = 30 To 35
    SWDS(i) = SWDS(i - 1) - EK(i - 5)
  Next
  SWDS(36) = SWDS(35)
  SWDS(37) = SWDS(36) - EK(31)
  SWDS(38) = SWDS(37) - EK(32)
  SWDS(39) = SWDS(38) - EK(33)
  SWDS(40) = SWDS(39) - EK(34)
  'Kapalı Kesit İlave Terim
  ReDim SW(1 To 1)
  For i = 2 To 6
    SW(1) = SW(1) + 2 * SWDS(i) * pl(i - 1) / (3 * pt(i - 1))
  Next
  SW(1) = SW(1) + 2 * SWDS(7) * pl(7 - 1) / (3 * pt(7 - 1))
  For i = 8 To 12
    SW(1) = SW(1) + 2 * 4 * SWDS(i) * pl(i - 1) / (3 * pt(i - 1))
  Next
  For i = 13 To 18
    SW(1) = SW(1) + 2 * SWDS(i) * pl(i - 2) / (3 * pt(i - 2))
  Next

```

```

SW(1) = SW(1) + SWDS(19) * pl(19 - 3) / (3 * pt(19 - 3))
SW(1) = SW(1) / DSBTT

For i = 1 To 44
  If (i >= 20 And i <= 23) Or (i >= 41 And i <= 44) Then
    SWDS(i) = SWDS(i)
  Else
    SWDS(i) = SWDS(i) - SW(1)
  End If
Next

ElseIf ktip = 5 Then
  '*** Çift Gözlü Kapalı Kesit ***
  'Primer Kayma Gerilmesi Hesaplanacak Noktalarda Kesit Kalınlıkları
  For i = 1 To ktn
    If i >= 1 And i <= 5 Then TJ(i) = pt(i)
    If i >= 6 And i <= 12 Then TJ(i) = pt(i - 1)
    If i >= 13 And i <= 18 Then TJ(i) = pt(i - 2)
    If i >= 19 And i <= 25 Then TJ(i) = pt(i - 3)
    If i >= 26 And i <= 29 Then TJ(i) = pt(i - 4)
    If i >= 30 And i <= 35 Then TJ(i) = pt(i - 5)
    If i >= 36 And i <= 42 Then TJ(i) = pt(i - 6)
    If i >= 43 And i <= 48 Then TJ(i) = pt(i - 7)
    If i >= 49 And i <= 52 Then TJ(i) = pt(i - 8)
    If i = 53 Then TJ(i) = pt(i - 9)
  Next
  'Sekonder Kayma Gerilmesi Bulunacak Noktaların FW Çarpılma
  'Yüzeylerinin Hesaplanması
  SWDS(1) = 0
  For i = 2 To 6
    SWDS(i) = SWDS(i - 1) + EK(i - 1)
  Next
  SWDS(30) = 0
  For i = 29 To 27 Step -1
    SWDS(i) = SWDS(i + 1) - EK(i - 4)
  Next
  SWDS(7) = SWDS(6) - SWDS(27)
  For i = 8 To 13
    SWDS(i) = SWDS(i - 1) + EK(i - 2)
  Next
  SWDS(14) = SWDS(13)
  For i = 15 To 19
    SWDS(i) = SWDS(i - 1) + EK(i - 3)
  Next
  SWDS(26) = 0
  For i = 25 To 20 Step -1
    SWDS(i) = SWDS(i + 1) - EK(i - 3)
  Next
  'Kesitin Simetrisi
  SWDS(31) = 0
  For i = 32 To 36
    SWDS(i) = SWDS(i - 1) - EK(i - 6)
  Next
  SWDS(53) = 0
  For i = 52 To 50 Step -1
    SWDS(i) = SWDS(i + 1) + EK(i - 8)
  Next
  SWDS(37) = SWDS(36) - SWDS(50)
  For i = 38 To 43
    SWDS(i) = SWDS(i - 1) - EK(i - 7)
  Next
  SWDS(44) = SWDS(43)

```

```

For i = 45 To 49
    SWDS(i) = SWDS(i - 1) - EK(i - 8)
Next

'Kapalı Kesit İlave Terim
ReDim SW(1 To 2)
For j = 1 To IG
    i = 1
    SW(j) = SW(j) + (CSG(j, 1) / 3 * (SWDS(i) + 4 * (SWDS(i + 1) +
    SWDS(i + 2) + SWDS(i + 3) + SWDS(i + 4)) + SWDS(i + 5))) / TG(j,
    i)
    i = 7
    SW(j) = SW(j) + (CSG(j, 1) / 3 * (SWDS(i) + 4 * (SWDS(i + 1) +
    SWDS(i + 2) + SWDS(i + 3) + SWDS(i + 4)) + SWDS(i + 5))) / TG(j,
    i)
    i = 14
    SW(j) = SW(j) + (CSG(j, 1) / 3 * (SWDS(i) + 4 * (SWDS(i + 1) +
    SWDS(i + 2) + SWDS(i + 3) + SWDS(i + 4)) + SWDS(i + 5))) / TG(j,
    i)
    i = 20
    SW(j) = SW(j) + (CSG(j, 1) / 3 * (SWDS(i) + 4 * (SWDS(i + 1) +
    SWDS(i + 2) + SWDS(i + 3) + SWDS(i + 4)) + SWDS(i + 5))) / TG(j,
    i)
Next
'Lineer Denklem Takımını Çöz
Call denklem_cozum_carpilma

For i = 1 To ktn
    If i <= 19 Then
        SWIK(i) = FWI(1)
    ElseIf i >= 20 And i <= 26 Then
        SWIK(i) = FWI(1) - FWI(2)
    ElseIf i >= 27 And i <= 30 Then
        SWIK(i) = 0
    ElseIf i >= 31 And i <= 49 Then
        SWIK(i) = FWI(2)
    ElseIf i >= 50 And i <= 53 Then
        SWIK(i) = 0
    End If
    SWDS(i) = SWDS(i) - SWIK(i)
Next

End If 'Kesit Tipi
End Sub

```

```

Public Sub denklem_cozum_carpilma()
    'Katsayılar matrisi KM nin tersinin alınması
    Dim AraMatris() As Double
    Dim MatTersA() As Double
    Dim FrTersHes() As Double
    Dim ElemanSayisi As Integer

    ElemanSayisi = 1
    ReDim AraMatris(ElemanSayisi + 1, ElemanSayisi, ElemanSayisi)
    ReDim MatTersA(ElemanSayisi + 1, ElemanSayisi, ElemanSayisi)
    ReDim FrTersHes(ElemanSayisi, ElemanSayisi, ElemanSayisi)
    ReDim FrTers(ElemanSayisi, ElemanSayisi)

```

```

For i = 0 To ElemanSayisi
  For j = 0 To ElemanSayisi
    AraMatris(0, i, j) = FR(i + 1, j + 1)
  Next
Next

For i = 0 To ElemanSayisi
  For j = 0 To ElemanSayisi
    MatTersA(i, j, i) = -AraMatris(i, j, i)
  Next
  n = 0
  Do While n <= ElemanSayisi
    MatTersA(i, n, n) = AraMatris(i, i, i)
    n = n + 1
  Loop
  MatTersA(i, i, i) = 1
  For j = 0 To ElemanSayisi
    For n = 0 To ElemanSayisi
      For m = 0 To ElemanSayisi
        AraMatris(i + 1, j, n) = AraMatris(i + 1, j, n) + MatTersA(i,
          j, m) * AraMatris(i, m, n)
      Next
    Next
  Next
Next
Next 'Denklemin sonu
'Son adımda elde edilen matrisinin tersinin bulunması
For i = 0 To ElemanSayisi
  MatTersA(ElemanSayisi + 1, i, i) = 1 / AraMatris(ElemanSayisi + 1, i,
    i)
Next

'KM Matrisinin
'A2*A1
For i = 0 To ElemanSayisi
  For n = 0 To ElemanSayisi
    For j = 0 To ElemanSayisi
      FrTersHes(0, i, n) = FrTersHes(0, i, n) + MatTersA(1, i, j) *
        MatTersA(0, j, n)
    Next
  Next
Next

Next
'Diğer Çarpımlar
For l = 1 To ElemanSayisi 'MTERS matrisinden bir düşük satırda
  For n = 0 To ElemanSayisi
    For i = 0 To ElemanSayisi
      For j = 0 To ElemanSayisi
        FrTersHes(l, i, n) = FrTersHes(l, i, n) + MatTersA(l + 1, i, j) *
          FrTersHes(l - 1, j, n)
      Next
    Next
  Next
Next
Next

'Ters Matrisin KMTERS matrisine aktarılması
For i = 0 To ElemanSayisi
  For j = 0 To ElemanSayisi
    FrTers(i, j) = FrTersHes(ElemanSayisi, i, j)
  Next
Next

```

```

For i = 0 To ElemanSayisi
  FWI(i + 1) = 0
  For j = 0 To ElemanSayisi
    FWI(i + 1) = FWI(i + 1) + FrTers(i, j) * SW(j + 1)
  Next
Next
End Sub

```

```

Public Sub kesit_gerilme()
  'Kesit içindeki Gerilmelerin Bulunması
  Dim sayac_2 As Integer
  ReDim MBS(aciklik - 1, Pny)
  ReDim MBP(aciklik - 1, Pny)
  ReDim MB(aciklik - 1, Pny)
  ReDim BM(aciklik - 1, Pny)
  ReDim FIM(aciklik - 1, Pny)
  ReDim SIGMA(aciklik - 1, Pny, kns - 1)
  ReDim TOPP(aciklik - 1, Pny, ktn - 1)
  ReDim TOSK(aciklik - 1, Pny, ktn - 1)

  For i = 0 To aciklik - 1
    If Ytip(i) = 3 Then sayac_2 = Pny
    If Ytip(i) <> 3 Then sayac_2 = Pn
    For j = 0 To sayac_2
      MBS(i, j) = V(i, j, 3) - V(i, j, 1) 'SEKONDER BURLMA
      MOMENTI
      MBP(i, j) = V(i, j, 1) 'PRIMER BURULMA MOMENTI
      MB(i, j) = V(i, j, 3) 'TOPLAM BURULMA MOMENTI
      BM(i, j) = V(i, j, 2) 'BİMOMENT
      FIM(i, j) = V(i, j, 0) / (G * Jb) 'KESİT DÖNME AÇISI
      '* Normal Gerilmesinin Hesaplanması *
      For m = 0 To kns - 1
        SIGMA(i, j, m) = BM(i, j) / FWW * W(m + 1)
      Next
      '* Primer Kayma Gerilmesinin Hesaplanması *
      If RO = 1 Then
        'Açık kesit
        For m = 0 To ktn - 1
          TOPP(i, j, m) = MBP(i, j) * TJ(m + 1) / Jb
        Next
      Else
        'Kapalı Kesit
        For m = 0 To ktn - 1
          TOPP(i, j, m) = MBP(i, j) * FIBT(m + 1) / Jb
        Next
      End If
      '* Sekonder Kayma Gerilmesinin Hesaplanması *
      For m = 0 To ktn - 1
        TOSK(i, j, m) = -MBS(i, j) * SWDS(m + 1) / (FWW * TJ(m + 1))
      Next
    Next
  Next
End Sub

```

**ÖZGEÇMİŞ**

Doğum tarihi 27.08.1980

Doğum yeri İstanbul

Lise 1991-1997 Sarıyer İH Lisesi

Lisans 1997-2001 İstanbul Üniversitesi Mühendislik Fak.  
İnşaat Mühendisliği Bölümü

Yüksek Lisans 2002-2005 Yıldız Teknik Üniversitesi Fen Bilimleri Enstitüsü  
İnşaat Müh. Anabilim Dalı, Yapı Programı

**Çalıştığı kurum(lar)**

2003-Halen Baran Yapı Müh. ve Müş. İnş. San. ve Tic. Ltd. Şti.

