

176621

**MATLAB® PROGRAMI DİLİNDE
MİKRODALGA ARAÇ KUTUSU HAZIRLANMASI**

MEHMET SAİT VURAL

**Mersin Üniversitesi
Fen Bilimleri Enstitüsü**

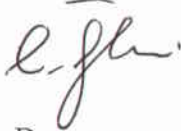
**Elektrik-Elektronik Mühendisliği
Ana Bilim Dalı**

YÜKSEK LİSANS TEZİ

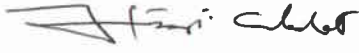
**Tez Danışmanı
Doç. Dr. Caner ÖZDEMİR**

**MERSİN
HAZİRAN – 2005**

Bu tezin gerek bilimsel içerik, gerekse elde edilen sonuçlar açısından tüm gerekleri sağladığı kanaatine ulaşan ve aşağıda imzaları bulunan biz jüri üyeleri, sunulan tezi oy birliği ile Yüksek Lisans Tezi olarak kabul ediyoruz.



Tez Danışmanı
Doç. Dr. Caner ÖZDEMİR



Jüri Üyesi
Yrd. Doç. Dr. Hüseyin CANBOLAT



Jüri Üyesi
Yrd. Doç. Dr. Hüseyin MUTLU

Bu tezin Fen Bilimleri Enstitüsü yazım kurallarına uygun olarak yazıldığı Enstitü Yönetim Kurulu'nun 15.08.2005 tarih ve 2005.21/365 sayılı kararıyla onaylanmıştır.



Prof. Dr. Mahir TURHAN
Enstitü Müdürü

Not: Bu tezde kullanılan özgün bilgiler, şekil, çizelge ve fotoğraflardan kaynak göstermeden alıntı yapmak 5846 sayılı Fikir ve Sanat Eserleri Kanunu hükümlerine tabidir.

ÖZ

Bu tez “Mikrodalga teorisi” için geliştirilmiş olan ve görsel bir platformda programlanan bir araç kutusudur. Hazırlanan araç kutusu içerisinde farklı konular bulunmaktadır. Bunlar İletim Hattı Teorisi, İletim Hattı ve Dalga Klavuzu, Mikrodalga Ağ Analizi, Mikrodalga Çınlayıcı, Empedans Uyumlama ve Frekans Uyumlama şeklinde konular ve bu konulara ait problemler kullanıcıya görsel bir platformda sunulmaktadır.

Yapılan program ile kullanıcı MATLAB’da herhangi bir fonksiyon veya bir fonksiyona ait herhangi bir parametre bilmeden programa ait fonksiyonun içeriğini hatasız ve doğru bir biçimde çalıştırarak programı görsel bir alanda kullanabilmektedir.

Hazırlanan araç kutusu MATLAB’ın bize sunduğu GUI (Graphical User Interface–Grafiksel Kullanıcı Arayüzü) teknolojisinden yararlanılmıştır ve görsel elemanlar diğer yüksek seviyeli diller olan Visual Basic, C ve Visual C++ programları tarafından GUI içerisine eklenilmiştir. Böylece Araç kutusu içerisindeki arayüz tanımlamalarından veya örnek animasyonlarda daha kolay ve kullanışlı bir kullanım sağlanmıştır.

ANAHTAR KELİMELEER : Mikrodalga Teorisi, Grafiksel Programlama

ABSTRACT

This thesis introduces a new and Visual Microwave MATLAB toolbox that we have developed for Microwave field theory. In this toolbox, different topics including Transmission line theory, Transmission lines and Waveguides, Microwave Network analysis, Microwave Resonators, Impedance matching and tuning and so on that have been transferred to visual medium to offer a simpler usage capability to the end user.

With this construct, the user does not need to know the MATLAB basics since he or she will only enter the parameter values visually.

The toolbox has been put into MATLAB GUI (Graphical User Interface) by applying some visual components in Visual Basic, C and Visual C++ computer languages. Different simulations and animations are given as examples to better describe the usage of the toolbox.

KEY WORDS : Microwave Theory, Graphical Programming

TEŐEKKÜR

Bu tez alıőmamda yardımını esirgemeyen danıőman hocam Do. Dr. Caner Özdemir'e , katkılarından dolayı Yrd. Do. Dr. Hüseyin Canbolat'a ve bugünlere gelmemde en büyük emeėi olan maddi manevi hiç bir zaman beni yalnız bırakmayan, desteklerini esirgemeyen babam ve anneme teőekkür eder bu tezimi babam ve anneme armaėan ederim.

İÇİNDEKİLER DİZİNİ

	<u>SAYFA</u>
ÖZ	i
ABSTRACT	ii
TEŞEKKÜR	iii
İÇİNDEKİLER DİZİNİ	iv
ŞEKİLLER DİZİNİ	viii
ÇİZGELER DİZİNİ	x
SİMGE VE KISALTMALAR DİZİNİ	xi
1. GİRİŞ	1
2. KAYNAK ARAŞTIRMALARI	5
2.1. MATLAB PROGRAMI	6
2.1.1. Matlab Derleyicisi	6
2.1.2. Matlab Programının Dezavantajları	6
2.1.3. Windows İşletim Sistemi için Gereksinimler	7
2.2. MATLAB PROGRAMLAMADA GUI(GRAFİKSEL ARABİRİM) TASARIMI	7
2.2.1. Matlab GUI ve GUIDE	8
2.2.2. Matlab’da Kod Oluşturarak GUI Oluşturma	13
2.2.2.1. Object Handles (Nesne Göstergesi)	18
2.3. MATLAB’DA NESNESEL DİĞER PROGRAMLARIN KULLANIMI	20
2.4. MİKRODALGA ARAÇ KUTUSU TASARIMI	21

2.5. MİKRODALGA ARAÇ KUTUSU TASARIMI İÇİN MİKRODALGA TEOREMLERİ	21
2.5.1. İletim Hattı Problemi	22
2.5.1.1. Koaksiyel iletim hattı problemi	22
2.5.1.2. İletim hattı katsayısı	23
2.5.1.3. İletim hattı empedansı	24
2.5.1.4. İletim hattı karakteristik empedansı (Açık Devre için)	24
2.5.1.5. İletim hattı karakteristik empedansı (Kapalı Devre için) ...	25
2.5.1.6. İletim hattı yansıma katsayısı	26
2.5.1.7. İletim hattı yitimsiz hat empedansı	26
2.5.1.8. İletim hattı bakış açısı empedansı	27
2.5.1.9. İletim hattı duran dalga oranı	28
2.5.2. TE Modundaki Dikdörtgensel Dalga Kılavuzunda Kompleks Yayılm Sabiti	28
2.5.3. TE Modundaki Dairesel Dalga Klavuzunda Kesim Sıklığı Dalga Numarası	29
2.5.4. TE Modundaki Dairesel Dalga Klavuzunda Kılavuz Dalga Uzunluğu	29
2.5.5. TE Modundaki Dairesel Dalga Klavuzunda Evre Hızı	30
2.5.6. TE Modundaki Yayılm Sabiti	31
2.5.7. TE Modundaki Dairesel Dalga Kılavuzundaki Dalga Empedansı	31
2.5.8. TE Modundaki Dairesel Dalga Klavuzunda Kılavuz Dalga Uzunluğu	32
2.5.9. TM Modundaki Dairesel Dalga Klavuzunda Kompleks Yayılma Sabiti	33
2.5.10. TM Modundaki Dairesel Dalga Klavuzunda Kesim Sıklığı Dalga Numarası	33
2.5.11. TM Modundaki Dairesel Dalga Klavuzunda Kılavuz Dalga Uzunluğu	34

2.5.12. TM Modundaki Dairesel Dalga Klavuzunda Evre Hızı	35
2.5.13. TM Modundaki Yayılım Sabiti	35
2.5.14. TM Modundaki Dairesel Dalga Klavuzundaki Dalga Empedansı	36
2.5.15. TM Modundaki Dairesel Dalga Klavuzunda Klavuz Dalga Uzunluğu	37
2.5.16. TM Modundaki Dairesel Dalga Klavuzunda Evre Hızı	37
2.5.17. TE Modundaki Yayılım Sabiti	38
2.5.18. TM Modundaki Dairesel Dalga Klavuzundaki Dalga Empedansı	39
2.5.19. TM Modundaki Dairesel Dalga Klavuzunda Klavuz Dalga Uzunluğu	39
2.5.20. Koaksiyel Kesim Sıklığı Problemi	40
2.5.21. Koaksiyel Yayılım Sabiti Problemi	40
2.6. GÖRSEL MİKRODALGA ARAÇ KUTUSU TASARIMI VERSION	
1.0	41
3. MATERYAL ve METOT	42
3.1. MATLAB'DA .M DOSYALARININ OLUŞTURULMASI	42
3.2. MATLAB GUI KULLANIMI	43
3.2.1. GUI Oluşturma için Temel Bileşenler	43
3.2.2. GUI Örnek Uygulamalar	58
3.2.2.1. Örnek uygulama düğme oluşturma	59
3.2.2.2. Örnek uygulama dosya çağırma ve disk işlemleri	59
3.2.2.3. Örnek uygulama - nesnel grup dizenegi	61
3.2.2.4. Örnek uygulama uipanel ve toolbar hazırlama	62
3.2.2.5. Örnek uygulama parent ve child proseslere erişim	64
3.2.2.6. Örnek uygulama matlabda hareketli animasyonu	64

3.2.2.7. Örnek uygulama frame (çerçeve)	65
3.2.2.8. Örnek uygulama istenilen değere göre işlem	68
3.2.2.9. Örnek uygulama slider (kayan çubuk)	69
3.2.3. GUI İle Web Bağlantısı	70
3.2.4. External Dosya Bağlantısı	70
3.3. GÖRSEL PROGRAMLAMA	91
3.3.1. Visual Basic ile Bağlantı	91
3.3.2. Visual Basic Script ve Java Script Kullanımı	99
3.3.3. Xml Formatında Veri Oluşturma	101
3.4. DERLEME	101
4. BULGULAR ve TARTIŞMA	104
5. SONUÇ ve ÖNERİLER	116
5.1. SONUÇLAR	116
5.2. ÖNERİLER	117
KAYNAKLAR	119

ŞEKİLLER DİZİNİ

ŞEKİL	SAYFA
Şekil 2.1. Guide Ekran Görüntüsü	8
Şekil 2.2. Guide Ekran Görüntüsü –Tasarım	9
Şekil 2.3. Özellikler Penceresi	9
Şekil 2.4. GUI Genel Yapısı	19
Şekil 3.1. Hareketli GUI Yayılım Sabit Animasyonu	43
Şekil 3.2. GUIDE Başlangıç Ekranı	44
Şekil 3.3. GUI Ana Yapısı	44
Şekil 3.4. Popup Menu GUI-Tasarım	46
Şekil 3.5. GUI Renk Seçenekleri –Dialog penceresi ile	48
Şekil 3.6. GUI Düğme oluşturma ve Callback	49
Şekil 3.7. GUI Düğme oluşturma ve Callback – Hareketli grafik	50
Şekil 3.8. CloseRequest ile Figure Sonlandırma	53
Şekil 3.9. Nested ile Text ve Pushbutton kullanımı	54
Şekil 3.10. Nesnelerle Tasarım	57
Şekil 3.11. Nesne ve Menü Kullanımı	57
Şekil 3.12. Pushbutton oluşturma	59
Şekil 3.13. Mikrodalga Gözet Penceresi	60
Şekil 3.14. Mikrodalga Araç Kutusu yol ayarı	61
Şekil 3.15. Seçeneğe göre İşlem yapma	62
Şekil 3.16. Araç Kutusu Tasarlama	63
Şekil 3.17. Parent-Child Erişim	64
Şekil 3.18. Hakkında Paneli – Tasarımı	65
Şekil 3.19. Hakkında Paneli	65
Şekil 3.20. Frekans Ölçüm Animasyonu	67
Şekil 3.21. Slider Animasyonu.....	69
Şekil 4.1. Normal Çalışır Ekran görüntüsü	105
Şekil 4.2. TM Moddaki simülasyona bağlantı	105
Şekil 4.3. Dikdörtgensel dalga uzunluğu simülasyonu	106
Şekil 4.4. Örnek web uygulaması	107
Şekil 4.5. Dalga kılavuzu araç kutusunun hesapladığı kesim frekans değerleri...108	

Şekil 4.6. Değerleri girilen dalga kılavuzu sunulan alan animasyon	109
Şekil 4.7. Dairesel dalga klavuzunda TM11 modu elektrik alan animasyonu	114
Şekil 4.8. Dikdörtgenel dalga klavuzunda TE20 elektrik alan Animasyonu	115
Şekil 4.9. Dikdörtgenel dalga klavuzunda TM11 3 Boyutlu animasyonu	115

ÇİZELGELER DİZİNİ

ÇİZELGE	SAYFA
Çizelge 2.1. GUI Temel Komutlar	12
Çizelge 2.2. GUI Olay Yordamları	13
Çizelge 2.3. Object Pascal nesne oluşturma.....	15
Çizelge 3.1. GUI Renk Seçenekleri	48
Çizelge 3.2. Eksen özellikler	50
Çizelge 3.3. Çizgi özellikler.....	50
Çizelge 3.4. İşaret özellikler.....	52
Çizelge 3.5. Mcc compiler seçenekleri.....	103
Çizelge 4.1. Dikdörtgensel dalga kılavuzunda TE ve TM modları için m ve n değerlerine göre kesim frekans değerleri	108

SİMGE VE KISALTMALAR DİZİNİ

η	:	Dalga empedansı
η_0	:	Görelî Geçirgenlik
ϵ	:	Görelî izinlik
a	:	İç yarıçap
b	:	Dış yarıçap
Z_0	:	Karakteristik Empedansı
Z_I	:	Geri iletim hattının giriş empedansı
T	:	İletim Hattı Katsayısı
R	:	Yansıma Katsayısı
Z_{in}	:	İletim Hattı giriş empedansı
Fre	:	Frekans
Uzunluk	:	Ardışıl iki maksimum voltaj arasındaki mesafe
λ	:	Dalga Boyu
c	:	Hız
R_l	:	Yükleme direnci
R_L	:	Yansıma Katsayısı
SWR	:	İletim Hattı duran dalga oranı
η_r	:	Görelî Geçirgenlik
ϵ_r	:	Görelî izinlik
ω	:	Açısal Hız
K	:	Kompleks Yayılım Sabiti
K_c	:	Kesim Dalga numarası
λ_g	:	Klavuz dalga uzunluğu
v_p	:	Evre Hızı
β	:	Yayılım Sabiti
Z_{te}	:	Dalga Empedansı
M	:	Geçiş hesaplaması
λ	:	Dairesel Dalga Empedansı
F_c	:	Koaksiyel Kesim Sıklık Frekansı

1. GİRİŞ

Son yıllarda, kişisel bilgisayarların hem hız hem de hafıza olarak kapasitelerinin çok çabuk bir şekilde artması ve buna paralel olarak bilgisayarların programlanabilme işlevliklerinin gelişmesi, Elektromanyetik problemlerinde rahatlıkla bilgisayar ortamında benzetimlerinin yapılmasına olanak sağlamıştır [1, 2]. Mikrodalga problemleri çözen benzer programlar da mevcuttur [3].

Ancak, Elektrik Mühendislerinin araştırmalarında çok sık kullandıkları bir yazılım olan MATLAB 'da bu konuda bir araç kutusu henüz geliştirilememiştir. Bu tezde, MATLAB ortamında "görsel" olarak geliştirdiğimiz "MATLAB Görsel Mikrodalga Araç Kutusu" tasarımı amaçlanmaktadır [4].

MATLAB Grafiksel Kullanıcı Arayüzü (GUI), MATLAB programcısı tarafından hazırlanan grafik tabanlı uygulamaların, son kullanıcıya fare ve klavye arabirimi ile enteraktif olarak hitap etmesini sağlayan bir platformdur [5-7].

MATLAB GUI uygulamalarının gerekliliğinin temel sebeplerinin başında günümüzde hazırlanan uygulamaların grafik tabanlı oluşu ve bu uygulamaların son kullanıcı tarafından kullanım kolaylığına sahip olması gelmektedir [4-7]. Bu sebeplerden dolayı, hazırlanan araç kutusunda MATLAB GUI (Kullanıcı ara yüzü) kullanılarak; "*Düğme*", "*Metin Kutuları*", "*Etiket*" ve "*Mönüler*" kullanılarak görsel ortam programına işlevlik kazandırılmıştır.

Geliştirilen mikrodalga araç kutusunun grafiksel arabiriminin hazırlanmasında, bir görüntüleme ekranı, birden fazla şekil içerebilir bir yapıda ve her şekil ise bir veya daha fazla '*uimenu*' ve '*uicontrol*' içerecek şekilde tasarlanmıştır

Uicontrol ile MATLAB genel yapısı içerisine yeni nesnelere katılmıştır. Bu nesnelere aslında fiğüre tabanlı uygun kodlar ile oluşturulan nesnelere dir. Komut düğmeleri, metin giriş kutuları, etiketler gibi. *Uimenu* ise bunların bir dizi şeklinde fiğüratif olarak belirlenmesini sağlar ve yine kodlarla oluşan bir veya birden fazla nesne için ayrı bir yordam oluşturmaktadır. Bunu karşımıza çıkan bir açılır liste kutusuna örnek olarak verebiliriz. Temel olarak MATLAB'da nesnelere bu şekilde oluşturularak istenilen Mikrodalga araç kutusu için alt yapıımız oluşturulmuştur.

Oluşturulan görsel mikrodalga araç kutusunun geliştirilmesi ise beş farklı aşamada yapılmıştır:

- 1) Tasarımlar ilk önce *Visual Basic* , *Visual C++* ve *MATLAB GUI* baz alınarak hazırlanmıştır. Buralarda hazırlanan özel benzetimler *MATLAB GUI* içerisine çalıştırılabilir *.exe formatında dönüştürülmüştür.
- 2) Daha sonra mönülerin seçeneklerinde kullanacak ve Mikrodalga konularının değişik denklemlerini çözen *MATLAB *.m* dosyaları hazırlanmıştır.
- 3) Ardından, bu M-dosyalarında oluşacak şekiller ve *MATLAB GUI* görsel kodları yazılmıştır.
- 4) Bunun ardından, mikrodalga elemanlarında fiziksel olayların animasyonları hazırlanmıştır. Örneğin, bir dalga klavuzunda elektrik ve manyetik alanların elektrik ve manyetik alanların çizilmesi ve zamana göre bu alanların ilerlemesi gibi animasyonlar yapılmıştır.
- 5) Son aşamada ise, geliştirilen araç kutusunun kullanıcı tarafından nasıl etkin kullanılacağı anlatıldığı ve *.pdf, *.html formatlarında hazırlanmış "Yardım" mönüsü programla ilişkilendirilmiştir.

Burada *.bat dosyalarıyla *MATLAB* (Normal *MATLAB* içerisinde) komut sistemi üzerinden yardım dosya yönlendirmeleri oluşturulmuştur. Böylece, hazırlanan araç kutusunun hem *MATLAB* hemde görsel ortamda çalışması sağlanmıştır.

Bu tezin esas amacı komutsal bir programlama dili olan *MATLAB* programının bu komutsal düzenekteki durumundan çıkartıp daha çok kullanıcı tabanlı ve görsel bir platforma yazılımı taşımaktır. Bunun esas nedenlerini şu şekilde sıralayabiliriz.

MATLAB'da programlar M-dosyalar şeklinde fonksiyonel bir yapı içerisinde hazırlanırlar (Programlar bu görsel yazılıma ek olarak Normal *MATLAB* içerisinde çalışacak şekilde bir klasörde fonksiyonel bir biçimde ilave olarak hazırlanmıştır). Bu M-dosya içerisinde konu ve konuyla ilgili yapılan işlemler ancak o programı yazan yazılımcıya ait algoritmik komutsal ifadelerdir. Hatta konu ile

ilgilenen kiři bile kullanılmıř olan teorem veya hesaplamayı bildiđi halde problemi çözmek için “Normal MATLAB” editörü içerisinde fonksiyonla ilgili olarak kaç parametre gireceđini veya hangi parametrenin ilk hangisinin son olacađını bilmede problem yařayabilecektir.

Bir ikincisi ise fonksiyonda bazen iřlemler dođru olsa bile ekrana sonu ile ilgili bilgiler gelmemektedir veya girilmesi gereken bir parametreye yanlıř veri giriřleri yapılabilmede ve kullanıcı uyarılmamaktadır.

Yukarıdaki problemleri gözönüne alarak yapılan hesaplamaların algoritmik yapıları olduđu gibi korunmuř ve kullanıcıya yapılan iřlem ile ilgili görsel bir arayüzde kullanılacak olan hesaplama ile ilgili bir platform oluřturulmuřtur. Burada hangi parametrenin kullanılacađını bilmeden kullanıcıya parametrelerin ne oldukları ile ilgili bilgi verilmiř ve o parametre deđerleri için yanıda giriř kutuları sunularak her kullanıcı için bir kolay düzene oluřturulmuřtur.

Bunun yanıda metin giriř kutularından birine bilgi giriři yapılmadıđı takdirde kullanıcıya uyarı mesajları ile doldurması gereken alanların olduđu uyarısı gelecek ve kullanılan hesaplamayla ilgili konu hakkında temel bilgiyede bu platform üzerinden ulařtırılabilecektir. Böylelikle yapılan hesaplamanın ne olduđu, ne için yapıldıđı ve nasıl bir sonu elde edileceđi ile ilgili dökümanlar sunularak konu ile ilgili bilgilerde programda sunulmaktadır.

Kullanılan yazılım herkesin rahatlıkla kullanacađı ve Mikrodalga teorisi içerisinde yer alan problemlere çözümler sađlayan bir simülasyon programı haline getirilmiř olmaktadır. Bunun yanıda “eđitim” içerikli bir yazılım olduđundan dolayı üniversitelerin ilgili bölümlerinde kullanılan bir program olma özelliđinide koruyacaktır.

Bu kullanım ile arayüzler geliřtirilerek diđer programlama dilleri gibi animasyonlar, simülasyonlar ve özel içerikli algoritmik ifadelerin bu programlama üzerinde kullanıcıya sunulması gerekleřtirilmiřtir ve normal MATLAB programının tamamıyla görsel bir platforma tařınmasının temelleri atılmıřtır.

Programın alıřma řekli ise, yazılım bize iki aılıř opsiyonu sunacaktır. ilk seçenekte klasik MATLAB editörüne geiř yaparak, hazırlanan M-dosyaların kullanımı sađlanmıřtır. Burada az da olsa, kullanıcıya görsel mesaj kutuları ile o anki mevcut dosyalar ve konularla ilgili yardımlar veya klasörlerde hangi bilgilerin

olduguna dair ipuçları da verilmektedir. Ancak bu kullanımın fonksiyonel bir çalışma yapısı bulunmamakta ve Visual Basic Scriptler, Java Scriptler, Visual Basic ,Visual C++ ve diğer .COM bileşenleri kullanamayacaktır. Bunun yanında; ikinci seçenek olan “*Windows MATLAB*” seçilirse bu arayüzlerin hepsi kullanılacaktır.

Görsel ortamda ise, program nesne tabanı desteği sağladığından dolayı, araç kutusunun kullanımı çok fonksiyonel hale gelmiştir. Bu ortamda, kullanıcının MATLAB programlama dilini bilmesi de gerekmemektedir. Çünkü, kullanılacak olan M-dosyaları görsel mönüler şeklinde, içlerindeki gereken parametre sorgularıyla beraber kullanıcıya sunulmaktadır. Kullanıcının sadece bu parametreleri uygun değerlerle doldurup, gerekli düğmelere basması yeterli olacaktır.

Program aynı zamanda *Visual Basic Script* ve *Java Script*'ler ile bir açılış sayfası şeklinde açılacaktır.

2. KAYNAK ARAŞTIRMALARI

Yapılan bu tez Mikrodalga teoremleri ve MATLAB 6.5 Programı faydalanılarak yapılmıştır [8-10]. MATLAB programının nesneye yönelik uygulaması olan GUI (Graphical Unit Interface-Kullanıcı ara birimi) – Grafikselsel arabirim’den yararlanılarak görsel dillerlede desteklenmiştir. Özellikle programlamada kullanılan ve normal MATLAB içerisinde bile nesnelere ve menüleri oluşturmamıza yarayan “*uicontrol*” ve “*uimönü*” komutlarının GUI ile etkileşimiyle yazılım oluşturulmaktadır.

Bu çalışmaya başlamadan önce elektromanyetik ve GUI üzerine çalışmalarını incelediğimizde, GUI (Kullanıcı ara birimi) için 2004 yılında Young Jin Choi, ve ark. [11] tarafından yapılan bir çalışmada “*Tomoğrafik*” görüntüleri işlemede kullanıcı arabirimi kullanılmış fakat sadece MATLAB komutsal düzeneği içerisinde istenilen hesaplamalar için çizilen grafiklerde nesnelere taban düşünülmeden sadece o grafik ile ilgili düğmeler yerleştirilmiştir.[6]

Bunun neticesinde istenilen işlem gerçekleşir gerçekleşmez tekrar MATLAB editörüne geri dönülerek GUI sadece çizim esnasında devrede olmaktadır. Bunun yanında kullanıcı arabiriminin kullanımı dışında gerek kullanıcıya görsel diğer etkenleri sunmak gerekse bazı metodsalsel işlemleri kolaylaştırmak adına “*Visual Basic Script*” veya “*Java Script*” kullanımı olmamıştır. Sadece bu kullanım Visual Basic olarak 2004’te Web tabanlı bir alet izleme sistemi adlı çalışmada kullanılmıştır [12]. Dikkat edilen husus çalışmaların tek bir ana hat içerisinde yapıldığı ve konunun sadece o yazılıma dayalı olduğudur. Web tabanlı sistemde yalnız Visual Basic kullanılmış iken diğer çalışmalarda sadece MATLAB kullanılmış bunun yanında mikrodalga için geliştirilen çok fazla araç kutusu bile bulunmamıştır.

I. M.-Baskir and A. V. Drozd [13] tarafından MATLAB’da dalga uzunluğu seçme ve analiz yöntemleri ile ilgili bir kullanıcı kullanıcı arayüzü geliştirmişlerdir. Yalnız mikrodalga kuramını sağlayan bu yapıda bile nesnelere özellikler tam olarak tümleşik bir yapı içerisinde çalışmamıştır. Yine buna benzer bir çalışma genetik bir toolbox için geliştirilmiştir [14].

Bütün bu çalışmalar incelendikten sonra özetle MATLAB’ın 6.5 versiyonu ile birlikte gelen GUIDE kullanılmadığı, figüratif kodlamanın sadece figür nesnesi

içerisinde tanımlandığı ve dışına çıkılmadığı, görsel programların tek başına değilde MATLAB ile birlikte kullanılmadığı ve en önemlisi bunların sadece MATLAB içerisinde bir klasörde tutulduğu ve derlenmediği görülmektedir.

Bu çalışmada yukarıdaki çalışmalardanda yola çıkarak bir çok yazılımı MATLAB içerisindeki bütün algoritmik ifadelerde ve diğer yazılımlarıda yaparak, bir simülasyon haline program getirilmiştir.[4-7]

2.1. MATLAB PROGRAMI

2.1.1. MATLAB Derleyicisi

MATLAB derleyicisi, MATLAB ile geliştirilen uygulamaları MATLAB'ın yüklü olmadığı sistemlerde çalıştırılabilir hale getirmektedir. MATLAB derleyicisi çalışılan işletim sistemine göre yürütülebilir uygulamalar oluşturmaktadır. Örnek olarak windows işletim sisteminde ".EXE" uygulamaları oluşturması gibi.kullanılan derleme programı "MATLAB Runtime Environment" ve "MCC" komut sistemidir. Bu sistemler ile .M dosyalar EXE'ye ve figürlerde .M dosyanın çalışmasında problem yansıtmayacak üzere Visual C++ desteği ile çalışır bir biçimde derlenmişlerdir.[6]

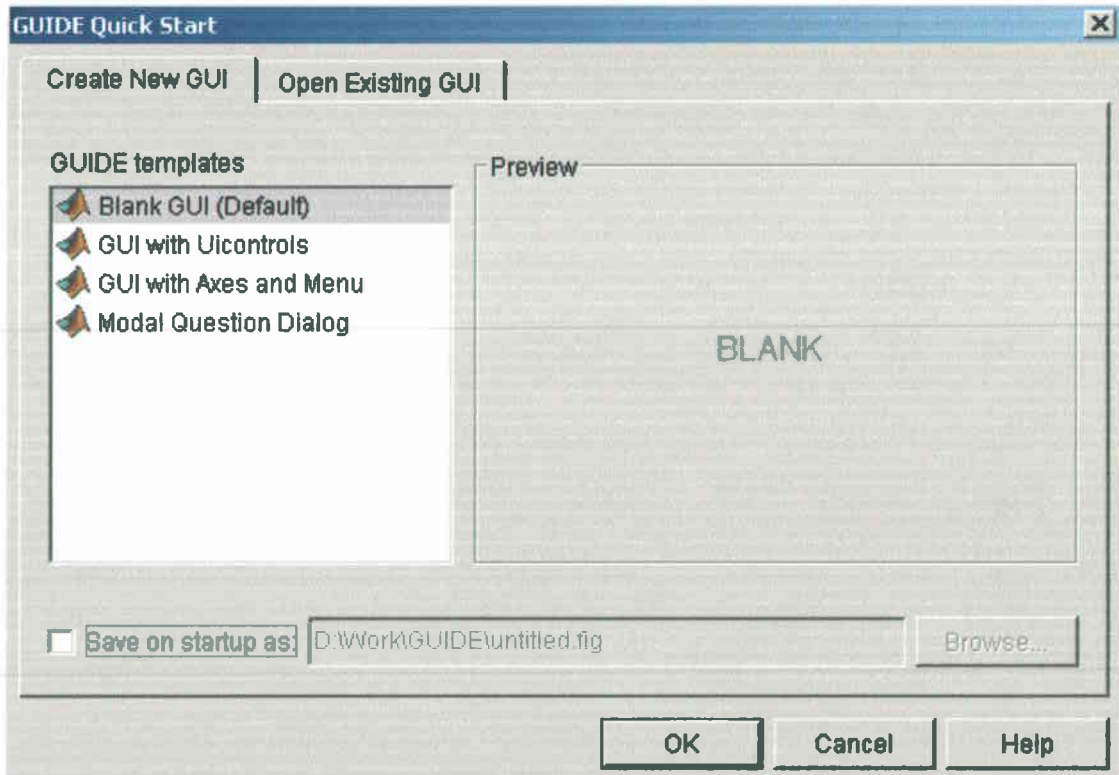
2.1.2. MATLAB Programının Dezavantajları

MATLAB uygulamaları diğer programlama dillerine göre daha geç açılmaktadır. Ama gelişen teknoloji ile bu fark gözardı edilebilmektedir. MATLAB 6.5 sürümünün çıkmasıyla bu dezavantaj gelişen teknolojiler sayesinde minimuma indirgenmiştir.

MATLAB, Basic, Fortran, C gibi programlama dilleri derleyicilerine nazaran beş veya on misli daha pahalıdır. Bu problem, farklı MATLAB paketleri çıkartılarak çözülmüştür. Böylelikle her kullanıcı kesimine hitap edebilmektedir. Öğrenciler için onlara uygun versiyon kurumlar için kurum işleyişine uygun paketler kullanılmaktadır [7].

2.2.1. MATLAB GUI ve GUIDE (Graphical Unit Interface-Grafiksel Ara Birim)

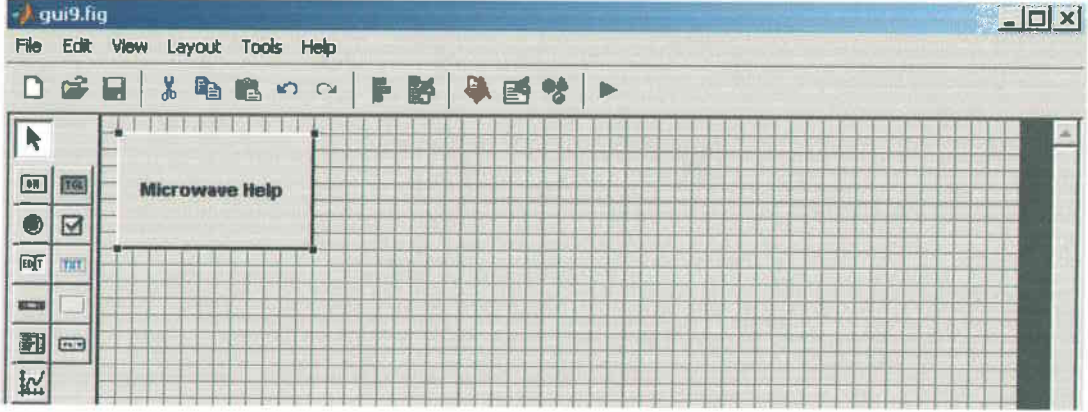
Bir GUI genelde iki türlü kullanılır birincisi tamamen nesneye yönelik kullanımlardır ki bu tür uygulamalarda MATLAB editöründe bilgilerin görüntülenmesi gerçekleştirilmez bile sadece dialog kutuları sayesinde ve direk olarak grafiksel birim içerisinde işlemler gerçekleştirilir. Diğeri ise text programlama içerisinde belli bölümlerde basit veri giriş /çıkışlarının ayarlamasında kullanılan kısımdır. Bu kısım belli komutsal ifadelerle oluşturulur. Aslında bütün nesnelere aynı kodlarla üretilmişlerdir [7, 8].



Şekil 2.1. GUIDE Ekran Görüntüsü

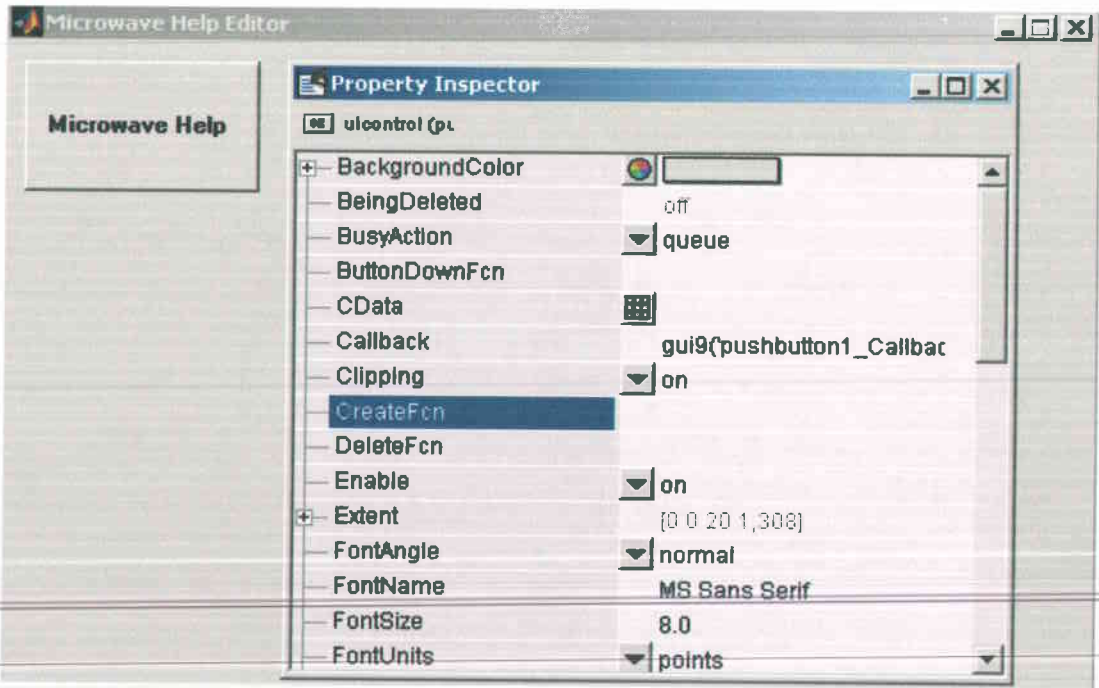
MATLAB Grafiksel arayüzü , kullanıcıya çalışması kolay bir alan hazırlar. Bu alan içerisinde Form; Tasarının yapılacağı yer, bu form üzerinde nesnelere –objeler, her bir nesneye ait “Özellikler” penceresi yer almaktadır [6]. Bu ifadelerin

değişimleri kodlarla yapılabildiği gibi programın başlangıç aşamasında buradanda ayarlanabilmektedir.



Şekil 2.2. Guide Ekran Görüntüsü -Tasarım

Nesne ve o nesneye ait özellikleri ayarlamak için ise Şekil 2.3.'teki figürden yararlanılır. Bu arada komut yordamları şu an için devrede olmayacağından yalnızca programsal olarak olayı başlangıç değer atama olarak ayarlanılmaktadır (initialize).



Şekil 2.3. Özellikler Penceresi

MATLAB GUI'nin içerisinde bulunan nesnellerimiz:

- Onay düğmesi – Push Button
- İki Yönlü Onay düğmesi – Toggle Button
- Tekli seçim düğmesi – Radio Button
- Çoklu seçim düğmesi – Checkbox
- Değiştirilebilir Metin Kutusu – Edit Box
- Sabit Metin Kutusu – Static Box
- Kaydırma Çubuğu – Slider
- Çerçeve – Frame
- Liste formlu çok seçenekli seçim Kutusu – listbox
- Açılımlı çok seçenekli seçim Kutusu – Pop-up mönü
- Eksenler – Axes

Bu nesnel sayesinde komut satırından klavye ile alınan bilgileri, grafiksel ortamda aldirmaya yarayan elemanlardır. Tabii bu işlemleri yaparken giriş ve çıkış fonksiyonlarıyla da bilgileri alıp sonuçlar üretilebilmektedir [6].

MATLAB GUI, nesnelere ve mönülerin bir arada tutulduğu bir yüzeydir. GUI yüzeyi kısaca elemanların dağılımını gösterir ve dağılımını tutar. Grafiksel yüzeyde dağılım göstergeler vasıtasıyla tutulmaktadır. Buna “handle point” de denilmektedir. Bir nesne oluşturulduktan sonra belirtece atanarak işlem görecektir. Bazı grafikler, örneğin; kontur çizimleri, bir çok nesneden ve bu nesneye ait belirteçlerden oluşmaktadır. Ana bir dizin ile ilgili işlem yapılıyorsa oda her zaman için “*Root*” (kök) yapısı içindeki belirteçtir ve değeri sıfırdır

```
get(0,'Screensize') veya  
set(0,'Background',[.7 .5 .1 ])
```

şeklinde değerler atanarak ilk başta karşımıza gelen figüre nesnesini kodlarla değiştirebiliriz.

Bir şeklin “figür” olarak belirteci, varsayılan olarak, bir tamsayıdır ve bu belirteç, şekil penceresinin başlık kısmında görüntülenir. Diğer nesne belirteçleri ise,

MATLAB tarafından kullanılan kayan-noktalı sayılardır. Dolayısıyla, bu belirteçleri kullanırken duyarlılıklarını tümüyle korumak gerekir. MATLAB'da bazı nesne belirteçlerine erişmek için gösterge fonksiyonlardan yararlanır. Bunlar;

- gcf – Şu anki figüre nesnesinin belirteci,
- gca – Şu anki axes (grafik) nesnesinin belirtecidir.

Bu fonksiyonlar, şekil ve eksen belirteçleri gerektiren diğer fonksiyonlara giriş argümanı olarak kullanılabilirler [6].

MATLAB M-dosya veya M-fonksiyon hazırlanırken yazılan komutlar yığını, GUI objelerinin yapacağı işlemler içinde belirtilmelidir. Bu işlemlere örnek olarak, kullanıcı bir düğmeye bastığında, istenen bir yerde tıklama işlemi sonucunda bir çıkış alınmasını sağlamak olabilir. Bu işlemlere program kod çağırılma yeri'(callback) de denilmektedir. Yapılan bu olay yordamları aynı şekilde programın kod kısmında bir M-dosyada oluşturulabilmektedir.

MATLAB GUI'de kullanılan temel komutlar Çizelge 2.1.'de verilmektedir.

Çizelge 2.1. GUI Temel Komutlar

<i>Komut, Değişken, Özellik</i>	<i>Açıklama</i>
<i>Gcbf</i>	Çalışmakta olan GUI Figürlü ifade edilmektedir
<i>Gebo</i>	Çalışmakta olan GUI Objesi ifade edilmektedir
<i>Persistent</i>	Sabir değer komutu
<i>Global</i>	Ortak değişken komutu
<i>Load</i>	Yükleme komutu
<i>Char</i>	Karaktere dönüştürme
<i>Str2num</i>	String bir ifadeyi nümerik bir ifadeye çevirir
<i>Num2str</i>	Nümerik bir ifadeyi String bir ifadeye çevirir
<i>Findobj</i>	Belirtilen özellikteki objeyi bulma komutu
<i>Set</i>	Verilen değeri özelliği belirtilen ifadeye atama
<i>Get</i>	Belirtilen nesnedeki değeri alma
<i>Tag</i>	Nesne için belirtilen etiket
<i>String</i>	Nesnenin adı
<i>Openfig</i>	M-dosya çağrılabilirlik şekli
<i>Handles</i>	Garfik nesnesinin göstergesi
<i>Loadimage</i>	Cdata ile oluşan imajı yükleme resim formatında
<i>Guihandles</i>	Figüre nesnesinin göstergesi atamak için

MATLAB programı içerisinde yeri geldiğinde işlemleri belli komut zincirlerinin işleyeceği şekilde ayarlamak mümkündür. Bunun yapılmasındaki temel unsur, oluşan nesnelere yönelik kodların yazılacağı durumdur. Fakat bu durum yerine göre değişebilmektedir. Örnek olarak tıklanıldığında farklı bir işlem yapacağı, farenin nesnenin üzerine gelindiğinde daha farklı işlemler gerçekleştirilebilmesidir. Bu durum gerçek uygulanabilir programlamanın OOP (Object Oriented Programming –Nesneye yönelik) olmasını sağlamıştır. Bu olay yordamlarından bazıları Çizelge 2.2.'de verilmektedir [4-7].

Çizelge 2.2. GUI Olay Yordamları

<i>Olay Yordamları</i>	<i>Açıklama</i>
<i>Callback</i>	Nesne Çağrıldığında / Nesneye tıklanıldığında çalıştırılacak işlev
<i>ButtonDownFcn</i>	Object basılı iken çalıştırılacak işlev
<i>CreateFcn</i>	GUI uygulaması başlatıldığında yapılacak işlev
<i>DeleteFcn</i>	Objeye kaldırıldığında yapılacak işlev
<i>ButtonDownFcn</i>	GUI yüzeyine fare ile tıklama sonucu oluşacak işlev
<i>CloseRequestFcn</i>	GUI penceresi kapatılırken oluşacak işlev
<i>CreateFcn</i>	GUI penceresi oluşturulurken oluşacak işlev
<i>DeleteFcn</i>	GUI yüzeyi kaldırılırken oluşacak işlev
<i>KeyPressFcn</i>	GUI yüzeyinde bir tuşa basıldığında oluşacak işlev
<i>ResizeFcn</i>	GUI yüzeyi boyutlandırılırken oluşacak işlev
<i>WindowButtonDownFcn</i>	GUI penceresinde aşağıya alma düğme işlevi
<i>WindowButtonUpFcn</i>	GUI penceresini bitiltme düğme işlevi
<i>WindowButtonMotionFcn</i>	GUI penceresinde Taşıma düğme işlevi

2.2.2. MATLAB’da Kod Oluşturarak GUI Oluşturma

MATLAB Grafikselsel arayüzü, kullanıcıya çalışması kolay bir alan hazırlar. Bu alan içerisinde nesnelere işlemleri gerçekleştirmek mümkündür. MATLAB 6.5 versiyonu ile bu çok kolay bir şekilde bize sunulmakta ve programcı bunu kullanarak istenilen işlemi gerçekleştirebilmektedir.

Programcı, yapması gereken bir hesaplamayı bir figür içerisinde yapmak istiyorsa ve bunu MATLAB’ın eski sürümlerinde gerçekleştirmek istiyorsa grafikselsel alanı kendi oluşturmalıdır. Çizilen bir grafiğin içerisinde sağ tıklayarak grafiğin renginin değişmesinin sağlanması gibi. Yalnız bu aşamaya gelmezden önceki sürümlerde GUI belli bir görsellik ve kod kullanımı dışında kalmaktaydı. Örnek olarak MATLAB 4 veya MATLAB 5’de böyle bir alan olmadığından programcılar nesneye yönelik bir programlama oluşturmak için nesneyi tanımlamalı ve buna

uygun olay yordamında kodlarla oluşturmak zorundadır [5-7]. Tıpkı Pascal programlamadaki “*Object*” veya C Programlamadaki “*Class*” ifadeler gibi, önce istenilen işlemler ayrı bir blokta tutulacak sonra grafiksel nasıl bir şekil isteniyorsa o çizilecek, farenin konumu belirlenecek, üzerine gelinmesi, fareye tıklanılması gibi sonrada hesaplamanın faresel işlemde sonra değerin bulunup başka bir grafiksel görüntüde ekrana yansımalarının sağlanması amaçlanacaktır. Pascaldaki nesneyi buna örnek vermek gerekirse; Çizelge 2.3.’teki gibi önce şekil çizilecek, fare aktif olacak ve hesaplama kod ile oluşturulacaktır [7].

Şeklinde ancak düğme oluşturulabilmekte ve oluşan düğmenin koordinatsal düzleminde gerekli nesneye ait kod çağırma yordamı (Callback) olay yordamı o an içerisinde yazılabilmektedir. İşte MATLAB bu kodlamaya gerek kalmadan C programlama dilinde yararlanarak derlenebilecek şekilde tasarlanmış kodlar oluşturmuş ve bu yolla nesnelere üretebilmiştir. Burada nesne belirteci (Object Handles) adı verilen ifadelerle ihtiyacımız olacaktır. Bu olayda bilgisayar ekranına yerleştirilen her bir figür tasarlanabilmektedir. Bu işlem ana başlangıç yerinden alt bitiş alanına kadar tekrarlanmaktadır (Parent-Child) [4].

Ayrıca, MATLAB ortamında herhangi bir görsel öğe kullanılmadığında mönü ('Propertyname') ile tanımlamalar yapılmıştır.

Örneğin,

Mönütasar = mönü ('Animasyon', 'Oklu Animasyon', 'Dikdörtgensel Dalga Klavuzu Animasyonu')

gibi sabit, ancak programlamaya görsel öğelerin katıldığı alt programlarda eklenebilmektedir. Altta gösterilen program parçasında olduğu gibi, 'Giriş', 'Çıkış' adlı iki seçeneği mönü ortamında, görsel olarak ve öncelik sırası belirleme opsiyonunu da sunarak kontrol etmek mümkün olmaktadır [5].

```
sec1=uiMENU('Label','Giriş');
sec2=uiMENU('Label','Çıkış');
k=get(sec1,'position');
if k==1
    msgbox('Şu an 1. seçenektensiniz');
    set(sec1,'Position',2);
else
    set(sec1,'Position',1);
    set('duğma1','Checked','off');
end;
```

Aynı zamanda kullanılan bir mönü tasarımı içerisinde de mönünün seçilmesi durumunda yapacağı işlevi tanımlayabiliriz. Bunu yapmak için aşağıda gösterildiği üzere 'Callback' komutu kullanılmıştır.

```
Ust=uimenu ( 'Label','Ornek' );  
Alt=uimenu(ust,'Label','Axis ','  
        'Callback',[
```

Görüldüğü gibi 'Uimönü' nesnelere kullanıcı arabirim nesnelere olup; şekil pencerelerinde mönü tasarlamak için kullanılırlar. Bunlar, şekil nesnelereinden türetilir ve eksenlerden bağımsızdırlar. 'Uicontrol' nesnelere ise kullanıcı arabirim nesnelere olup; kullanıcının seçtiği nesnelere üzerinde fonksiyonların nasıl işletileceklerini ayarlarlar [5]. Pascal programlama dilinin nesneye yönelik programlama özelliğinden de faydalanılarak, Visual C++'dan MATLAB GUI içerisine, nesnelere yerleştirilmiştir.

Geliştirilen mikrodalga araç kutusunun grafiksel arabiriminin hazırlanmasında, bir görüntüleme ekranı, birden fazla şekil içerebilir bir yapıda ve her şekil ise bir veya daha fazla 'uimönü' ve 'uicontrol' içerecek şekilde tasarlanmıştır.

GUI ile tasarlanan object MATLAB'da ise bu tamamen sınırlandırılabilir. Öyleki kullanıcının ne gireceğine sizin ekranda göstereceğiniz bir giriş kutusu (text) ve komut düğmesiyle (Pushbutton) hesaplanacak verinin başka bir Etiket (label) içerisinde gösterilmesini o konuyu yalnızca bilen bir kişinin değil herkesin kullanılabileceği bir şekilde tasarlanmıştır. Çünkü kullanıcının yapması gereken sadece metin kutularına verileri girip düğmeye tıklamaktır. Aynı zamanda *nargin* (Giriş parametre kontrolü) ve *nargout* (Çıkış parametre kontrolü) ilede parametereye göre algoritmik programın çalıştırılabilme olayını çok rahat bir biçimde burada yapabiliriz. Zaten bu yapının kullanılması Object programlama içerisinde olayı dinamik bir yapıya ulaştırmış olacaktır.

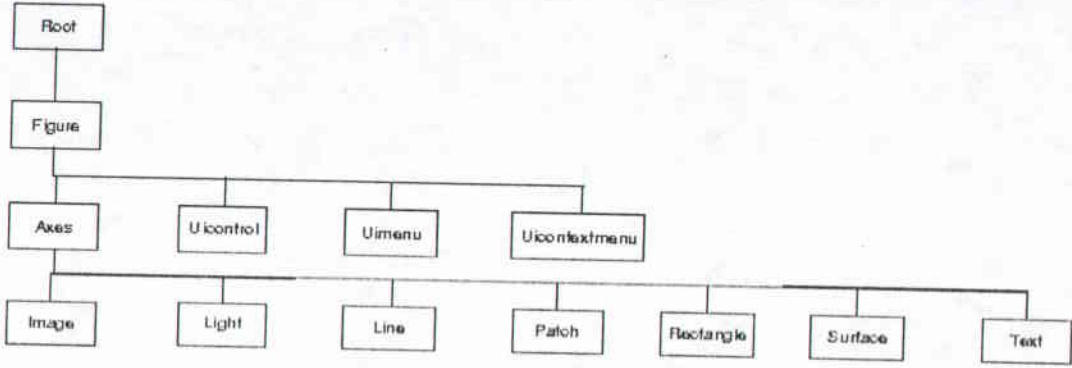
Bu olaya komut kontrol tabanlı programlama olan pascaldaki gibi düşünersek. "Object Pascal" kullanarak nesneyi tanımladıktan sonra nesne için tasarlanılan fonksiyonlar, prosedürler ve diğer elemanların yapacağı işler ve içerikler belirtildikten sonra grafik yüzeyinde yapılan değişikliklerle ekranda nesnelere gözükülecektir. Bu yapıda "Variant Record" adı verilen bir yapının tümleşik olarak tip (Type) blokları kullanılarak yapılan olayın MATLAB'daki nargin ile aynı ifadeyi sağlaması gerçekleştirilmektedir. Böylelikle en az kod ile çok daha kullanışlı fonksiyonların MATLAB'da kullanıldığını görülmektedir. MATLAB içerisindeki

GUI kullanılarak mnler ve dięer formlardan oluřan belli bir amaca uygun programlarda geliřtirilebilir. Tanımlanan yapılar da GUI ierisindeki her nesne Visual Basic'teki gibi hazır bir biimde olcađından sadece yordamına uygun kodları oluřturmamız yeterli olacaktır.

2.2.2.1. Nesne gstergeci (Object handles)

Kk nesnesinden itibaren iřlemlerin gerekleřmesi sađlanıyorsa burada figrlerle iřlemler esas alınacaktır. Yalnız bir programlamada izilen birden fazla figr ierisinden hangisi ile ilgili iřlem yapılacađına karar veren ve iřlem gren ifadenin gerekleřmesi iin o figr zerinde iřlemin gerekleřmesini sađlayan nesne gsterge (handles) komutlarına ihtiya vardır. Bylelikle bu gsterge komutu sayesinde iřlemler o figr zerinde tasarlanılmaktadır. Temel olarak bir grafik nesnesi zerinde tasarlanılan belli iřlemler vardır. Bunlar:

- Figr – Grafik ortamının izileceđi pencere
- Axes – Figr ierisindeki koordinatsal eksenler
- Line – Axes zerindeki veri noktaları
- Text – Axes koordinat dzlemi ierisine string ifade yazılması
- Umen – Figr penceresi ierisine programlanabilir mn tanımları oluřturma
- Uicontrol – Programlanabilir kullanıcı arayz ierisinde nesnelere oluřturulacak dzenek , textler, dğmeler, kaydırma ubukları,...



Şekil 2.4. GUI Genel Yapısı

bulunmaktadır. Bunlar temel yapı içerisinde en çok değinilen noktalar olmuştur. Çünkü yapı bu ifadelerle nesnelere oluşturmakta veya menüleri tasarlayabilmektedir. Hem sağ tıklama ile oluşacak mönü hemde sol tıklama ile oluşacak mönü ile tasarımlar gerçekleşir. Burada en önemli ifadeler “Uicontrol”, “Uimenu” ve “Uicontextmenu” komutlarıdır. Bunların temel kullanımları aşağıdadır. Bu komutlar yardımıyla düğmeler, nesnelere, mөнüler, mөнülerin içerikleri, popup mөнü yapıları oluşturulabilir. Genel kontrol komut kullanımları aşağıda verilmektedir.

I) Uicontextmenu

```
set(0,'DefaultUicontextmenuCreateFcn','set(gcf,"IntegerHandle","off")')
```

II) Uimenu Properties

```
set(0,'DefaultUimenuPropertyName',PropertyValue...)
```

```
set(gcf,'DefaultUimenuPropertyName',PropertyValue...)
```

```
set(menu_handle,'DefaultUimenuProperty',PropertyValue...)
```

III) Uicontrol Properties

```
set(0,'DefaultUicontrolProperty',PropertyValue...)
```

```
set(gcf,'DefaultUicontrolProperty',PropertyValue...)
```

Bu kontroller aşamasında düğmeler, metin kutuları (textler) ve diğer yapılar Guide komutu kullanmadan ayrı bir arayüzle ekrana getirilebilir. Örnek vermek

gerekirse bir düğmeyi kod ile oluşturma aşamasında “Uicontrol” komutu kullanılarak gerekli parametreler verildiği takdirde figür içerisinde yeni bir nesneyi verilen koordinatlara taşıyabilir ve nesneye yönelik işlemlerde yine komutsal düzenekte kullanabiliriz. “Ekrantemiz” adlı bir düğmeyi oluşturarak tıklanıldığı anda figür nesnesi içerisindeki tüm nesnelere ortadan kaldırabiliriz.

2.3. MATLAB’DA NESNESEL DİĞER PROGRAMLARIN KULLANIMI

MATLAB’da tasarlayacağımız mikrodalga araç kutusunun simülasyon ve arayüzlerinin kullanıcıya daha görsel ve kullanışlı bir halde sunmak için *Visual Basic*, *Visual C++*, kodlarıyla tümleşik olarak çalışacak şekilde programlar oluşturularak MATLAB’ın kullanımında bize kolaylık sağlayan ve MATLAB programcıları tarafından çok fazla kullanılmayan (“*External File*”) diğer dosya sistemleri özel hazırlanmış fonksiyonel komutlar içerisinde yazılarak sistem içerisine konulmuşlardır.

Bu yapılar GUI içerisine yerleştirilirken bazen sistem içine direk olarak entegre edebileceğimiz gibi bazı yerlerde *.EXE* olarak çağırarak figürde diğer *EXE* programı çalıştırırız. Yalnız bu kullanıma çok fazla yer verilmeyerek *Visual Basic* kodları GUI içerisine yerleştirilmiştir. *Visual C++* daha çok derleme aşamasında kullanılmıştır.

Bir nesneyi oluştururken hatalarla karşılaşıldığında bu hataları giderme veya görsel bir platformu dinamik bir yapı üzerinde tasarlarken bu tür kodlamalar çok işe yaramıştır. Öyleki programlamada MATLAB düzleminden bakıldığında yapılan yazılımlar *.m*, *.fig*, *.mat* formatlarında olmaktadır. Biz bunları bütün işletim sistemlerinde çalışacak bir düzeneğe çevirirken bile problemler yaşadık. Çünkü bir figür dosyası içerisinde bir mönüden başka bir figüre içerisindeki mönünün bir seçeneğinin çağırılması MATLAB’da olağan dışıydı. Bu nedenden hatalar ürettiyordu bunu çözmek için her birinin ayrı ayrı derlenmesi sağlandı ve *Visual C++* ile de tümleşik uygulama birleştirildi.

Programın işleyişi sırasında web programlamada yararlanılmıştır. Düzenekte sunulan “*Web yardım*” ile kullanıcıya *Visual Basic Script* ve *Java Script* kullanılarak web sayfası formatında dosyalarda hazırlanılmıştır. Bu sistem

“Windows MATLAB” için gerçekleştirilmiştir. Hatta programın belli yerlerinde veriler *XML* formatında çevrilebilmektedir. Fakat “*Normal MATLAB*” seçeneği seçilirse o zaman yardım (help) seçeneği ile *.BAT* ile oluşturulmuş olan yardım dosyamız ekrana gelecektir. Burada görüntülenme GUI’de olduğu gibi bir arayüzde gerçekleşmeyecektir. Gerçekleşecek düzenek normal MATLAB programıdır.

2.4. MİKRODALGA ARAÇ KUTUSU TASARIMI

Burada yapılan araştırmada görülmüştür ki programlama esas alındığında çeşitli araç kutuları tasarımı MATLAB’da yapıldığı halde görsel bir araç kutusu bulunmamaktadır.

Fakat Mikrodalga için çeşitli tasarımlar yapılmış ve buna uygun Mikrodalga mühendisliği için çeşitli fonksiyonel yapılar oluşturulmuştur. Yapılan tasarımlar sadece MATLAB içerisinde kalmakta ve çok az bir alan içerisinde, sadece simülasyonların bazıları *GUI* içerisinde programlanmış ve “Normal MATLAB” editörüne yine bağımlı kalarak MATLAB bilmeden program yine kullanılamayacaktır. Bunun yanında hiç *.COM* özelliği kullanılmamış ve diğer nesnel programlama dillerinde faydalanılmamıştır.

Hatta bu tezde sadece diğer nesnel programlar *.EXE* şeklinde çağrılmayıp bizzat programın gerekli yerlerinde çağrılarak işlem görmektedir. Yeri geldiğinde pascalda bir nesnenin çalışması için *.EXE* ‘ye çevrilmiş, C Programlama dilindeki karşılığı ile karşılaştırılıp her ikisi derleme (Compile) edilerek sonuçlar irdelenmiş ve *GUI* içerisinde diğer nesnelere oluşumlarına geçilmiştir. Tez içerisinde sadece animasyonlar değil programsal olarak bütün bir şekilde *EXE* çalışmaktadır.

2.5. MİKRODALGA ARAÇ KUTUSU TASARIMI İÇİN MİKRODALGA TEOREMLERİ

Burada mikrodalga araç kutusunda tasarlanılan hesaplamalar ve açıklamaları yer almaktadır. Aşağıda listesi verilen sistemlere özel problem ve animasyonlarda eklenmiştir. Genelde birçok formül ve hesaplama olduğundan Sadece İletim Hattı (Transmission Line) Problemi için kullanılacak formül ve hesaplamalar verilmiştir.

Diğer hesaplanacak hesaplamalarda yine verilen İletim Hattı problemi şeklinde tasarlanılmıştır. [15-21] Bu problemdende anlaşılacağı üzere fonksiyonel bir yapı için giriş parametreleri ve buna uygun hesaplanacak denklemler, formüller ve sonuçta bulunacak sonuç değişkenleri verilmektedir. Bu düzeneğin aynısı “Mikradalga” adlı MATLAB içerisinde hazırlanan klasör ilede sunulmaktadır. “Görsel MATLAB” ile bu değişkenler nesnelere atanarak işlemler gerçekleştirilmektedir.

2.5.1. İletim Hattı Problemi

İletim hattı probleminde temel olarak aşağıdaki konulardan oluşmaktadır. [3,18-20]

- Koaksiyel İletim hattı problemi (Coaxial Transmission Line)
- İletim Hattı Katsayısı (The Transmission Coefficient)
- İletim Hattı Empedansı (The input Impedance of the transmission line)
- İletim Hattının Karakteristik Empedansı (Açık Devre için) (Characteristic Impedance of the transmission line for open circuit)
- İletim Hattının Karakteristik Empedansı (Kapalı Devre için) (Characteristic Impedance of the transmission line for short circuit)
- İletim Hattı Yansıma Katsayısı (The Reflection Coefficient)
- İletim Hattı Yitimsiz Hat Empedansı (The input Impedance at a distance length from the load on a lossless line)
- İletim Hattı Bakış Açısı Empedansı (The input Impedance of the transmission line)
- İletim Hattı Duran dalga oranı (The Standing wave ratio on the line)

2.5.1.1. Koaksiyel iletim hattı problemi

Giriş Parametreleri (Input Parameters)

η_0 : Görelî Geçirgenlik (Relative Permeability)

ϵ : Görelî izinlik (Relative Permittivity)

- a : Koaksiyel hattın iç yarıçapı (Inner Radius of the coaxial line)
b : Koaksiyel hattın dış yarıçapı (Outer Radius of the coaxial line)

Çıkış Parametreleri (Output Parameters)

Z_o : Koaksiyel hattın Karakteristik Empedansı (The characteristic Impedance of the coaxial line)

$$\eta = \sqrt{\frac{\eta_o}{\epsilon}} \quad (2.1)$$

$$Z_o = \frac{\eta}{2\pi} \ln \frac{b}{a} \quad (2.2)$$

2.5.1.2. İletim hattı katsayısı

Giriş Parametreleri (Input Parameters)

Z_o : İletim hattının Karakteristik Empedansı (Characteristic Impedance of the transmission line)

Z_I : Geri iletim hattının giriş empedansı (The input Impedance seen by the feed line)

Çıkış Parametreleri (Output Parameters)

T : İletim Hattı Katsayısı (The Transmission Coefficient)

R : Yansıma Katsayısı (The Reflection Coefficient)

$$T = 1 + R \quad (2.3)$$

$$R = \frac{Z_1 - Z_0}{Z_1 + Z_0} \quad (2.4)$$

$$T = \frac{2Z_1}{Z_1 + Z_0} \quad (2.5)$$

2.5.1.3. İletim hattı empedansı

Giriş Parametreleri (Input Parameters)

Z_0 : İletim hattının Karakteristik Empedansı (Characteristic Impedance of the transmission line)

Z_1 : Geri hat giriş empedansı (The input Impedance seen by the feed line)

Çıkış Parametreleri (Output Parameters)

Z_{in} : İletim Hattı giriş empedansı (The input Impedance of the transmission line)

$$Z_{in} = \frac{Z_0^2}{Z_1} \quad (2.6)$$

2.5.1.4. İletim hattı karakteristik empedansı (açık devre için)

Giriş Parametreleri (Input Parameters)

Z_0 : İletim hattının Karakteristik Empedansı (Characteristic Impedance of the transmission line)

f_{re} : Frekans (Frequency)

Uzunluk (Length) : Ardışıl iki maksimum voltaj arasındaki mesafe (The distance between two successive voltage maxima(or minima))

Çıkış Parametreleri (Output Parameters)

Zin : İletim hattının giriş empedansı (The input Impedance of the transmission line)

Giriş Empedansı (I = 0 at the load ,as expected for an open circuit while the voltage is maximum-The input Impedance)

$$Z_{in} = -j Z_0 \cot(\beta \times \text{Uzunluk}) \quad (2.7)$$

2.5.1.5. İletim hattı karakteristik empedansı (kapalı devre için)

Giriş Parametreleri (Input Parameters)

Zo : İletim hattının Karakteristik Empedansı (Characteristic Impedance of the transmission line)

Fre : Frekans (Frequency)

Uzunluk (Length) : Ardışıl iki maksimum voltaj arasındaki mesafe (The distance between two successive voltage maxima(or minima))

Çıkış Parametreleri (Output Parameters)

Zin : İletim hattının giriş empedansı (The input Impedance of the transmission line)

Giriş Empedansı

$$Z_{in} = j Z_0 \tan(\beta \times \text{Uzunluk}) \quad (2.8)$$

Hesaplamalar için aşağıdaki diğer fonksiyonlara ihtiyacımız vardır.

$$c = 3 \times 10^8 \quad (2.9)$$

$$\lambda = \frac{c}{f_{re}} \quad (2.10)$$

$$\beta = \frac{2\pi}{\lambda} \quad (2.11)$$

$$Z_{in} = j Z_0 \tan(\beta \times \text{Uzunluk}) \quad (2.12)$$

2.5.1.6. İletim hattı yansımaya katsayısı

Giriş Parametreleri (Input Parameters)

Z_0 : İletim hattının Karakteristik Empedansı (Characteristic Impedance of the transmission line)

Z_1 : Geri hat giriş empedansı (The input Impedance seen by the feed line)

Çıkış Parametreleri (Output Parameters)

R : Yansımaya Katsayısı (The Reflection Coefficient input Impedance of the transmission line)

$$R = \frac{Z_1 - Z_0}{Z_1 + Z_0} \quad (2.13)$$

2.5.1.7. İletim hattı yitimsiz hat empedansı

Giriş Parametreleri (Input Parameters)

Z_0 : İletim hattının Karakteristik Empedansı (Characteristic Impedance of the transmission line)

Zl : Yükleme Empedansı (The Load Impedance)

Fre: Frekans (Frequency)

Uzunluk (Length) : Ardışıl iki maksimum voltaj arasındaki mesafe (The distance between two successive voltage maxima(or minima))

Çıkış Parametreleri (Output Parameters)

Zin : İletim hattının Mesafe giriş empedansı-Yüklemede empedansındaki ki uzunluk için (The input Impedance at a distance length from the load)

Giriş Empedansı :

$$Z_{in} = Z_0 \frac{Z_l + Z_0 \tanh(\gamma x \text{ Uzunluk})}{Z_0 + Z_l \tanh(\gamma x \text{ Uzunluk})} \quad (2.14)$$

2.5.1.8. İletim hattı bakış açısı empedansı

Giriş Parametreleri (Input Parameters)

Zo : İletim hattının Geri Karakteristik Empedansı (Characteristic Impedance of the transmission line)

Rl : Yükleme direnci

Zl : Yükleme Empedansı (The Load Impedance)

Fre: Frekans (Frequency)

Uzunluk (Length) : Ardışıl iki maksimum voltaj arasındaki mesafe (The distance between two successive voltage maxima(or minima))

Çıkış Parametreleri (Output Parameters)

Zin : İletim hattının giriş empedansı (The input Impedance of the transmission line)

$$Z_{in} = Z_l \frac{R_l + jZ_l \tan(\beta x \text{ Uzunluk})}{Z_l + jR_l \tan(\beta x \text{ Uzunluk})} \quad (2.15)$$

2.5.1.9. İletim hattı duran dalga oranı

Giriş Parametreleri (Input Parameters)

Z_0 : İletim hattının Karakteristik Empedansı (Characteristic Impedance of the transmission line)

Z_L : Yükleme empedansı (The Load Impedance)

Çıkış Parametreleri (Output Parameters)

R_L : Yansıma Katsayısı (The Reflection Coefficient of the transmission line)

SWR : İletim Hattı duran dalga oranı (The Standing wave ratio on the line)

$$SWR = \frac{1 + |R_L|}{1 - |R_L|} \quad (2.16)$$

2.5.2. TE Modundaki Dikdörtgenel Dalga Kılavuzunda Kompleks Yayılım Sabiti

Giriş Parametreleri (Input Parameters)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f : Frekans (Frequency)

ω : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

K : TE Modundaki Dikdörtgenel Dalga Kılavuzunda Kompleks Yayılım Sabiti (The Characteristic Impedance Of The Coaxial Line)

$$K = \omega \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.17)$$

2.5.3. TE Modundaki Dairesel Dalga Kılavuzunda Kesim Sıklığı Dalga Numarası

Giriş Parametreleri (Input Parameters)

n, m : İsteğe bağlı herhangi sayı düzeneği (any arbitrary number)

a : Dairesel Dalga kılavuzunun yarıçapı(Radius of the circular waveguide)

Çıkış Parametreleri (Output Parameters)

K_c : TE Modundaki Dairesel Dalga Kılavuzunda Kesim Dalga numarası
(The Cut off wavenumber of the TE Mode for circular waveguide)

$$K_c = \frac{R_{nm}}{a} \quad (2.18)$$

2.5.4. TE Modundaki Dairesel Dalga Kılavuzunda Kılavuz Dalga Uzunluğu

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f : Frekans (Frequency)

ω : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

λ_g : TE Modundaki Dairesel Dalga Kılavuzunda Klavuz dalga uzunluğu
(The Guide wavelength of the TE Modes for circular waveguide)

$$\beta = \sqrt{K^2 - K_c^2} \quad (2.19)$$

$$\lambda_g = \frac{2\pi}{\beta} \quad (2.20)$$

2.5.5. TE Modundaki Dairesel Dalga Kılavuzunda Evre Hızı

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (**Radius of the circular waveguide**)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f_{re} : Frekans (Frequency)

w : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

v_p : TE Modundaki Dairesel Dalga Kılavuzunda Evre Hızı (The phase velocity of the TE Modes for circular waveguide)

$$K = \omega \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.21)$$

$$\omega = 2 \pi f_{re} \quad (2.22)$$

$$\beta = \sqrt{K^2 - K_c^2} \quad (2.23)$$

$$v_p = \frac{\omega}{\beta} \quad (2.24)$$

2.5.6. TE Modundaki Yayılım Sabiti

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f : Frekans (Frequency)

ω : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

β : TE Modundaki Dairesel Dalga Kılavuzunda Yayılım Sabiti (The propagation constant of the TE Mode for circular waveguide)

$$\beta = \sqrt{K^2 - Kc^2} \quad (2.25)$$

2.5.7. TE Modundaki Dairesel Dalga Kılavuzundaki Dalga Empedansı

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f : Frekans (Frequency)

ω : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

Zte : TE Modundaki Dairesel Dalga Kılavuzunda Dalga Empedansı (The Wave Impedance Of The TE Modes For Circular Waveguide)

$$K = \omega \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.26)$$

$$\beta = \sqrt{K^2 - K_c^2} \quad (2.27)$$

$$M = 120\pi \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.28)$$

$$ZTE = \frac{KxM}{\beta} \quad (2.29)$$

2.5.8. TE Modundaki Dairesel Dalga Kılavuzunda Kılavuz Dalga Uzunluğu

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

Çıkış Parametreleri (Output Parameters)

λ : TE Modundaki Dairesel Dalga Empedansı (The Wave Impedance Of The TE Modes For Circular Waveguide)

$$\lambda = \frac{2\pi}{K_c} \quad (2.30)$$

2.5.9. TM Modundaki Dairesel Dalga Klavuzunda Kompleks Yayılma Sabiti

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f : Frekans (Frequency)

ω : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

K : TE Modundaki Dairesel Dalga Kılavuzunda Kompleks Yayılma Sabiti
(The Complex Propagation Constant Of The TM Mode For Circle Waveguide)

$$K = \omega \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.31)$$

2.5.10. TM Modundaki Dairesel Dalga Klavuzunda Kesim Sıklığı Dalga Numarası

Giriş Parametreleri (Input Parameters)

n, m : İsteğe bağılı herhangi sayı düzeneğı (any arbitrary number)

a : Dairesel Dalga kılavuzunun yarıçapı(Radius of the circular waveguide)

Çıkış Parametreleri (Output Parameters)

K_c : TM Modundaki Dairesel Dalga Kılavuzunda Kesim Dalga numarası
(The Cut off wavenumber of the TM Mode for circular waveguide)

$$K_c = \frac{Rnm}{a} \quad (2.32)$$

2.5.11. TM Modundaki Dairesel Dalga Kılavuzunda Kılavuz Dalga Uzunluğu

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f : Frekans (Frequency)

ω : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

λ_g : TM Modundaki Dairesel Dalga Kılavuzunda Kılavuz Dalga Uzunluğu
(The Guide Wavelength Of The TM Modes For Circular Waveguide)

$$\beta = \sqrt{K^2 - K_c^2} \quad (2.33)$$

$$\lambda_g = \frac{2\pi}{\beta} \quad (2.34)$$

2.5.12. TM Modundaki Dairesel Dalga Klavuzunda Evre Hızı

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f_{re} : Frekans (Frequency)

ω : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

v_p : TM Modundaki Dairesel Dalga Klavuzunda Evre Hızı (The Phase Velocity Of The TM Modes)

$$K = \omega \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.35)$$

$$\omega = 2 \pi f_{re} \quad (2.36)$$

$$\beta = \sqrt{K^2 - Kc^2} \quad (2.37)$$

$$v_p = \frac{\omega}{\beta} \quad (2.38)$$

2.5.13. TM Modundaki Yayılım Sabiti

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f_{re} : Frekans (Frequency)

ω : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

β : TM Modundaki Yayılım Sabiti (The Propogation Constant Of The TM Mode)

$$\beta = \sqrt{K^2 - Kc^2} \quad (2.39)$$

2.5.14. TM Modundaki Dairesel Dalga Kılavuzundaki Dalga Empedansı

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

f_{re} : Frekans (Frequency)

w : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

ZTE : TM Modundaki Dairesel Dalga Kılavuzundaki Dalga Empedansı (The Wave Impedance Of The TM Modes For Circular Waveguide)

$$K = \omega \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.40)$$

$$\beta = \sqrt{K^2 - Kc^2} \quad (2.41)$$

$$M = 120\pi \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.42)$$

$$ZTE = \frac{KxM}{\beta} \quad (2.43)$$

2.5.15. TM Modundaki Dairesel Dalga Klavuzunda Kılavuz Dalga Uzunluğu

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

Çıkış Parametreleri (Output Parameters)

λ : TM Modundaki Dairesel Dalga Empedansı (The Wave Impedance Of The TM Modes For Circular Waveguide)

$$\lambda = \frac{2\pi}{K_c} \quad (2.44)$$

2.5.16. TM Modundaki Dairesel Dalga Klavuzunda Evre Hızı

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

fre: Frekans (Frequency)

w : Açısıl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

v_p : TM Modundaki Dairesel Dalga Klavuzunda Evre Hızı (The Phase Velocity of The TM Modes)

$$K = \omega \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.45)$$

$$(2.46)$$

$$\omega = 2 \pi \text{fre}$$

$$\beta = \sqrt{K^2 - Kc^2} \quad (2.47)$$

$$v_p = \frac{\omega}{\beta} \quad (2.48)$$

2.5.17. TE Modundaki Yayılım Sabiti

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

fre: Frekans (Frequency)

w : Açısâl Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

β : TE Modundaki Yayılım Sabiti (The Propagation Constant of The TE Mode)

$$\beta = \sqrt{K^2 - Kc^2} \quad (2.49)$$

2.5.18. TM Modundaki Dairesel Dalga Kılavuzundaki Dalga Empedansı

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

η_r : Görelî Geçirgenlik (Relative Permeability)

ϵ_r : Görelî izinlik (Relative Permittivity)

fre : Frekans (Frequency)

w : Açısal Hız (Angular velocity)

Çıkış Parametreleri (Output Parameters)

ZTE : TM Modundaki Dairesel Dalga Kılavuzundaki Dalga Empedansı (The Wave Impedance Of The TM Modes For Circular Waveguide)

$$K = \omega \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.50)$$

$$\beta = \sqrt{K^2 - K_c^2} \quad (2.51)$$

$$M = 120\pi \sqrt{\frac{\eta_r}{\epsilon_r}} \quad (2.52)$$

$$ZTE = \frac{KxM}{\beta} \quad (2.53)$$

2.5.19. TM Modundaki Dairesel Dalga Kılavuzunda Kılavuz Dalga Uzunluğu

Giriş Parametreleri (Input Parameters)

a : Dairesel Dalga kılavuzunun yarıçapı (Radius of the circular waveguide)

Çıkış Parametreleri (Output Parameters)

λ : TM Modundaki Dairesel Dalga Empedansı (The Wave Impedance Of The TM Modes For Circular Waveguide)

$$\lambda = \frac{2\pi}{K_c} \quad (2.54)$$

2.5.20. Koaksiyel Kesim Sıklığı Problemi

Giriş Parametreleri (Input Parameters)

- a : İç Yarıçap(Inner Radius)
b : Dış Yarıçap (Outer Radius)
 ϵ_r : Göreli izinlik (Relative Permittivity)

Çıkış Parametreleri (Output Parameters)

F_c : Koaksiyel Kesim Sıklık Frekansı (Cut Off Frequency Of The Coaxline)

$$F_c = \frac{K_c}{2\pi\sqrt{\epsilon_r}} \quad (2.55)$$

2.5.21. Koaksiyel Yayılım Sabiti Problemi

Giriş Parametreleri (Input Parameters)

- a : İç yarıçap(Inner radius)
b : Dış Yarıçap (Outer radius)
 ϵ_r : Göreli izinlik (Relative Permittivity)

Çıkış Parametreleri (Output Parameters)

K_c : Koaksiyel Yayılım Sabiti Problemi (Propogation constant of the coaxline)

$$K_c = \frac{2}{a+b} \quad (2.56)$$

2.6. GÖRSEL MİKRODALGA ARAÇ KUTUSU TASARIMI VERSİON 1.0

Yapılan tez ile Mikrodalga mühendisliği için yararlı olabilecek bir program hazırlanarak sadece bir master tezi olarak değil üniversitelerin Elektrik Elektronik mühendisliği bölümlerinde kullanılacak bir simülasyon programı özelliğide katılmıştır.

Tezin esas amacı MATLAB programının özelliklerini diğer nesnel programlarla birleştirerek ortak uygulanabilir bir program oluşturmak ve bu program içerisinde *Mikrodalga* teoremlerini kullanmak olacaktır. Bir diğer nokta ise kullanıcı açısından, tez kullanıcının en az hata ile görsel bir platformda, yardım mөнüleriyle ve diğer işlemlerle Mikrodalga Mühendisliğine faydalı olabilecek bir yapı kazandırılması olacaktır. [21]

Yapılan yazılım ile program taşınabilirlik özelliğiyle tüm işletim sistemleri içerisinde çalışabilmektedir. MATLAB bilmeden program her tür kullanıcı kullanabilecektir. Bunun için grafiksel alandan faydalanılmış ve *GUI* komutlarıyla nesnel ve mөнüler oluşturularak görsel programlarla da arayüzler tasarlanılmıştır.

3. MATERYAL ve METOT

Tez daha çok programsal bir yapı üzerinde olduğundan temel olarak GUI (Grafiksel ara birim)'den bahsedilecektir. Bu tez çalışması için öncelikle mikrodalga hesaplamaları için m-dosyalar oluşturulmuş, oluşturulan bu m-dosyalar grafiksel bir alana taşınarak (GUI) hesaplamalar gerçekleştirilmiş. Bu arayüz diğer görsel programlar ile desteklendikten sonra bu görsel programlar içerisinde gerekli bağlantılar kurulmuştur. Tezin belki önemli ve karmaşık olan kısımlarından biri de yapılan programların tek bir programmış mantığıyla çalışıyor olmasıdır. İşte bu düzeniğin sağlanması için derlenmesine (compile) gerek duyulmuştur. Bu temel düzenek içerisinde GUI nasıl oluşmuş ve bu oluşan olayların kullanımları basit örneklendirmelerle verilerek buradan üretilenler “Bulgular ve Tartışma” bölümünde görsel şekillerle açıklanmıştır.

3.1. MATLAB'DA .M DOSYALARININ OLUŞTURULMASI

MATLAB programında m dosyalar ile her hesaplama ayrı bir dosyada tutulmaktadır. M-dosyanın hazırlanması MATLAB programında “*edit*” komutuyla gerçekleşmektedir. Bu editör sayesinde hesaplamak istediğimiz algoritmik ifade yazılarak çalıştırılır. Hazırlanan dosyalar fonksiyonel bir yapı içerisinde olup birbiri içerisinde programlar çağrılabilir. M-dosyalar birer fonksiyon şeklinde tanımlanmaktadır ve işlemler bu düzenekte korunmaktadır. Fonksiyonlar kullanım alanına göre parametre ile sınırlandırılmakta veya parametresiz sadece fonksiyon adına göre işlemektedir.

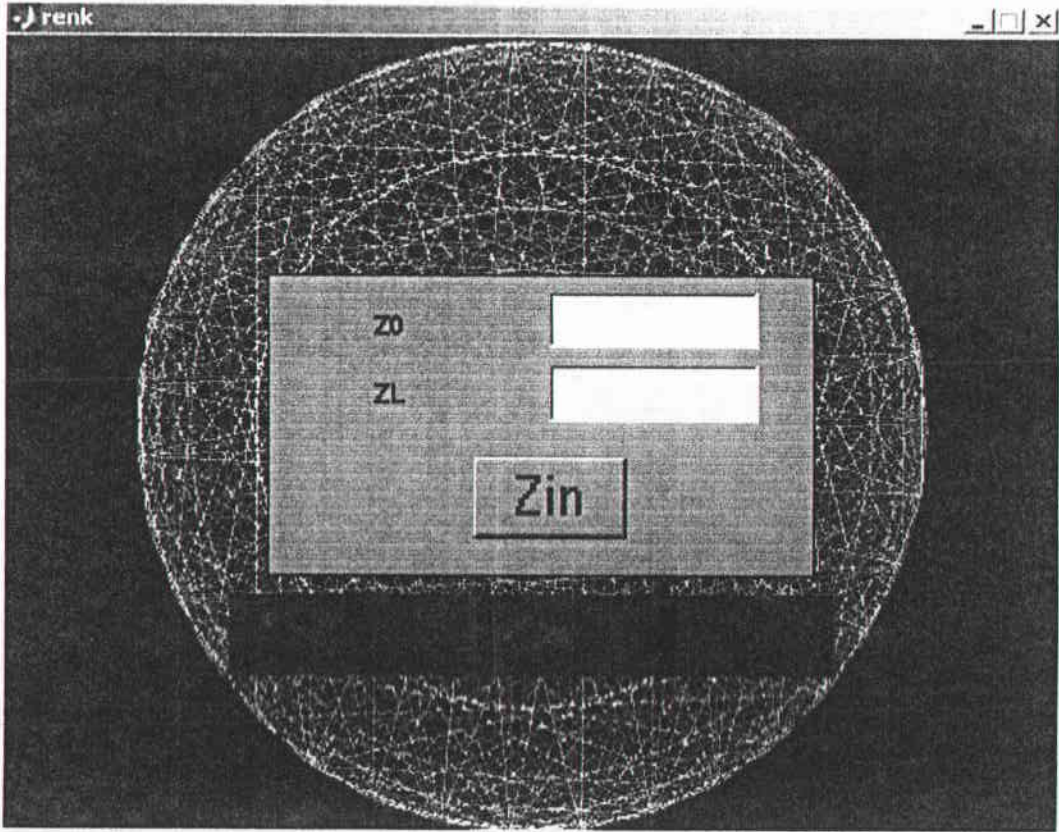
Mikrodalga araç kutusu tasarımında ister normal editör içerisinde yaptığımız mikrodalga klasörü içerisindeki hesaplamalar ister görsel programda oluşturduğumuz simülasyon programımız olsun esas temelde M-dosyalardan oluşmaktadır. Her yapılan işlem için ayrı bir M- dosya oluşturulmuştur. Guide ile oluşturulan bir m- dosya ile istenilen bir mikrodalga probleminin işlenmesi için m - dosya çağrılacaktır. Yani simülasyon programını dahi kullansak mikrodalga problemi içerisinde her hesaplama ayrı bir m-dosyada fonksiyonel bir yapıda tutulmaktadır.

3.2. MATLAB GUI KULLANIMI

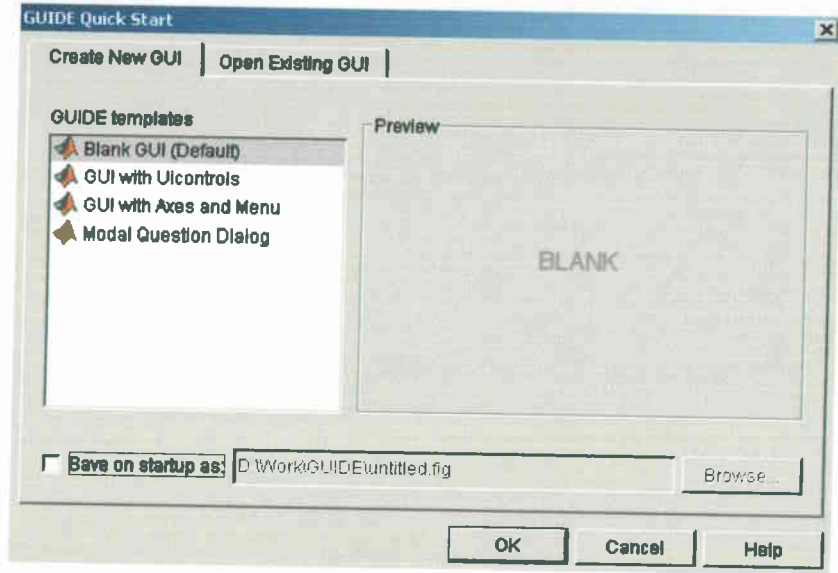
3.2.1. GUI Oluřturma için Temel Bileřenler

Bu kısımda GUI hakkında genel bir bilgi vererek yaptığımız programda oluşturduğumuz düğme , text , mönüler ... gibi ifadeleri kısaca nasıl oluşturduk kodları nasıl ayarladık bunları almadan yapı üzerinde biraz açıklama yapılacaktır.

Genelde herbir grup üzerinde tanımlamalar yapıldıktan sonra işlemlere geçilmiştir. Bu gruptan temel olarak Push Button, Static Text, ... üretilmektedir. Üretilen ifadeler Grup Nesneleri olarak bir figure içerisinde yer almaktadır.

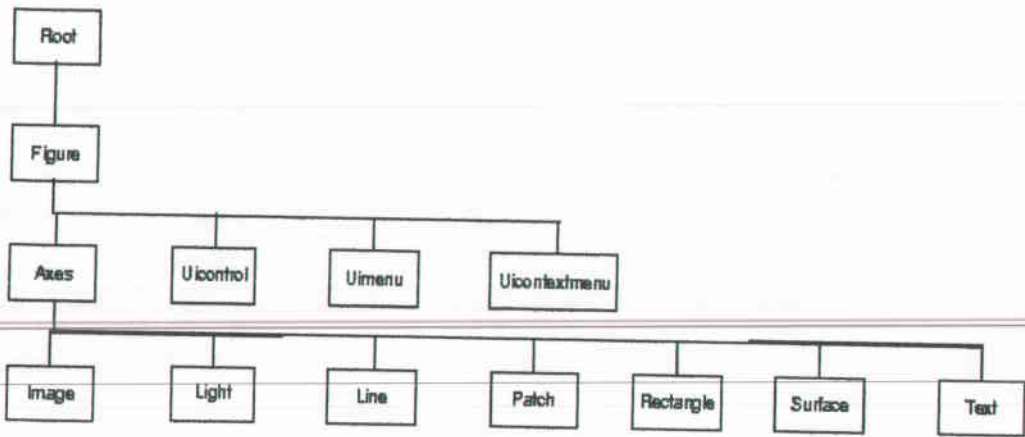


Şekil 3.1. Hareketli GUI Yayılım Sabit Animasyonu



Şekil 3.2. GUIDE Başlangıç Ekranı

Şekildende anlaşılacağı gibi dökümanlar oluşmaktadır. Bir GUI genelde iki türlü kullanılır. Birincisi tamamen nesneye yönelik kullanımlarda ki o tür uygulamalarda MATLAB editöründe bilgilerin görüntülenmesi gerçekleştirilmez bile sadece dialog kutuları sayesinde ve direk olarak grafiksel birim içerisinde işlemler gerçekleştirilir. Diğeri ise text programlama içerisinde belli bölümlerde basit veri giriş/çıkışlarının ayarlamasında kullanılan kısımdır. Bu kısım belli komutsal ifadelerle oluşturulur. Aslında bütün nesnelere aynı kodlarla üretilmişlerdir. Ondan dolayı genel diagram modeli aşağıda verilmiştir.



Şekil 3.3. GUI Ana Yapısı

Burada en önemli ifadeler `Uicontrol`, `Uimönü`, `Uicontextmönü` komutlarıdır. Bunların temel kullanımları aşağıdadır. Bu komutlar yardımıyla düğmeler, nesnelere, mönüler, mönülerin içerikleri, popüpmönü yapıları oluşturulabilir. Genel kullanımları aşağıda verilmektedir.

Uicontextmönü Properties

```
set(0,'DefaultUicontextmönüCreateFcn','set(gcf,"IntegerHandle","off")')
```

Uimönü Properties

```
set(0,'DefaultUimönüPropertyName',PropertyValue...)  
set(gcf,'DefaultUimönüPropertyName',PropertyValue...)  
set(mönü_handle,'DefaultUimönüProperty',PropertyValue...)
```

Uicontrol Properties

```
set(0,'DefaultUicontrolProperty',PropertyValue...)  
set(gcf,'DefaultUicontrolProperty',PropertyValue...)
```

Bu modeller içerisinde MATLAB'da nesnelere ve o nesneye ilişkin işlemler gerçekleştirilebilir. Root (kök) ana yapısından başlanılarak şekiller ve grafiklerle oluşturulan MATLAB programına yine arayüz (interface) olarak izin verilen aşamada eksenler(Axesler) ve diğer kontroller eklenebilir.

Bu kontroller aşamasında düğmeler textler ve diğer yapılar `Guide` komutu kullanmadan ayrı bir arayüzle ekrana getirilebilir. Örnek vermek gerekirse bir düğmeyi kod ile oluşturma aşamasında ***Uicontrol*** komutu kullanılarak gerekli parametreler verildiği takdirde `figure` içerisinde yeni bir nesneyi verilen koordinatlara taşıyabilir ve nesneye yönelik işlemlerde yine komutsal düzenekte kullanabiliriz. Ekranımız adlı bir düğmeyi oluşturarak tıklanıldığı anda `Figure` nesnesi içerisindeki tüm nesnelere ortadan kaldırmasını istiyorsak :

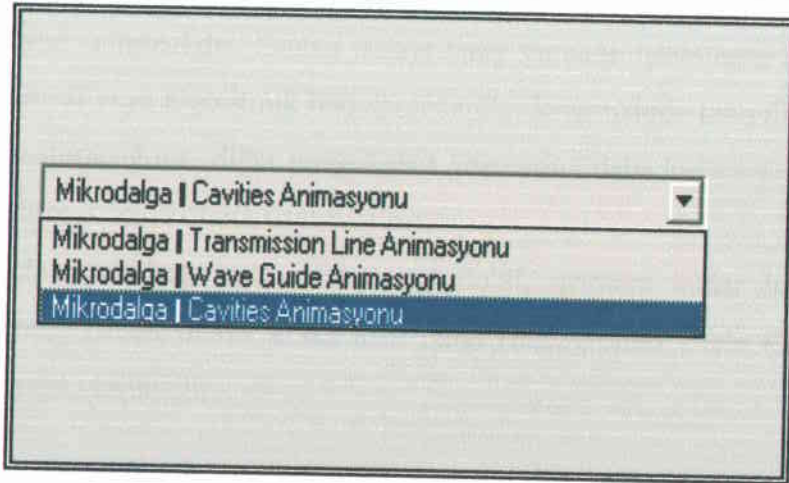
```
Ekranemiz = uicontrol('Style','Pushbutton','Position',...  
[20 150 100 70], 'Callback','cla','String','Temizle');
```

Bu sistem ile yapılan bir düğmenin koordinatsal düzlemine nesneyi yerleştirerek işlemlerimizleri gerçekleştirebiliriz. Bu nesnenin **Callback** işleviyle düğme, programsal açıdan bir algoritmik yapıyı oluşturacak ve düğmeye tıklayarak işlemleri gerçekleştirecektir.

Nasıl bir düğme oluşturulup tıklama olayı gerçekleşiyorsa buna benzer diğer nesnelere de bu yolla üretilen istediğimiz programları yazabiliriz. Bir **Popuımönü** örneğini yaparsak, burada *Combobox*'dan seçilen animasyonun grafiğini ekranda görmek esas amacımızdır. Bundan dolayı oluşturacağımız programda **Callback** yapan yordam içerisinde sadece grafik için komutu yazmamız gerekecektir.

```
hpopuımönü = uicontrol('Style','Popup','String',...
```

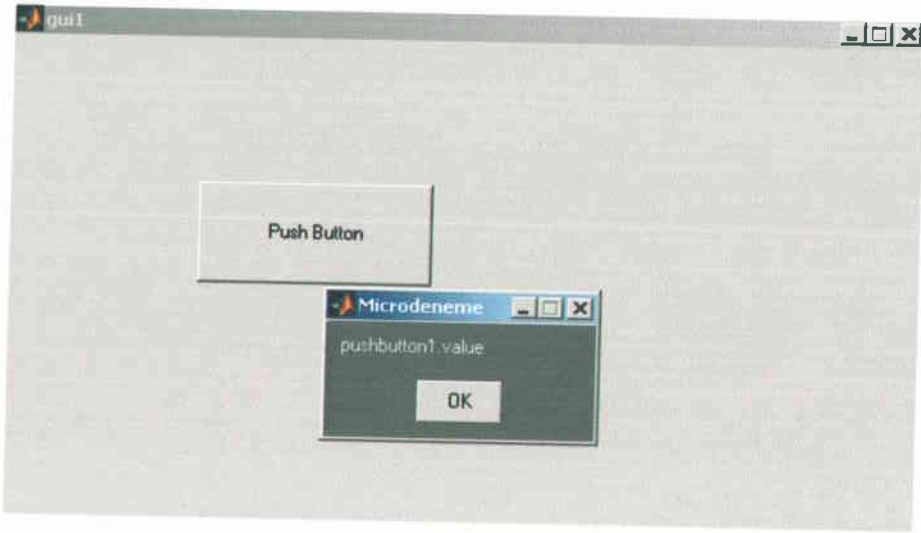
```
'Mikrodalga – Transmission Line Animasyonu | Mikrodalga – Wave Guide  
Animasyonu | Mikrodalga – Cavities Animasyonu ', 'Position',[20 320 100 50],...  
'Callback','Goruntu')
```



Şekil 3.4. Popuımönü GUI-Tasarı

Şeklinde verilen diagramlar ekranda görüntülenir. Burada Popuımönü mantığıyla işlemler gerçekleştirilir. Her üç seçenekte verilen koordinatsal düzlemde ayrı yordamlarda tutulmaktadır.

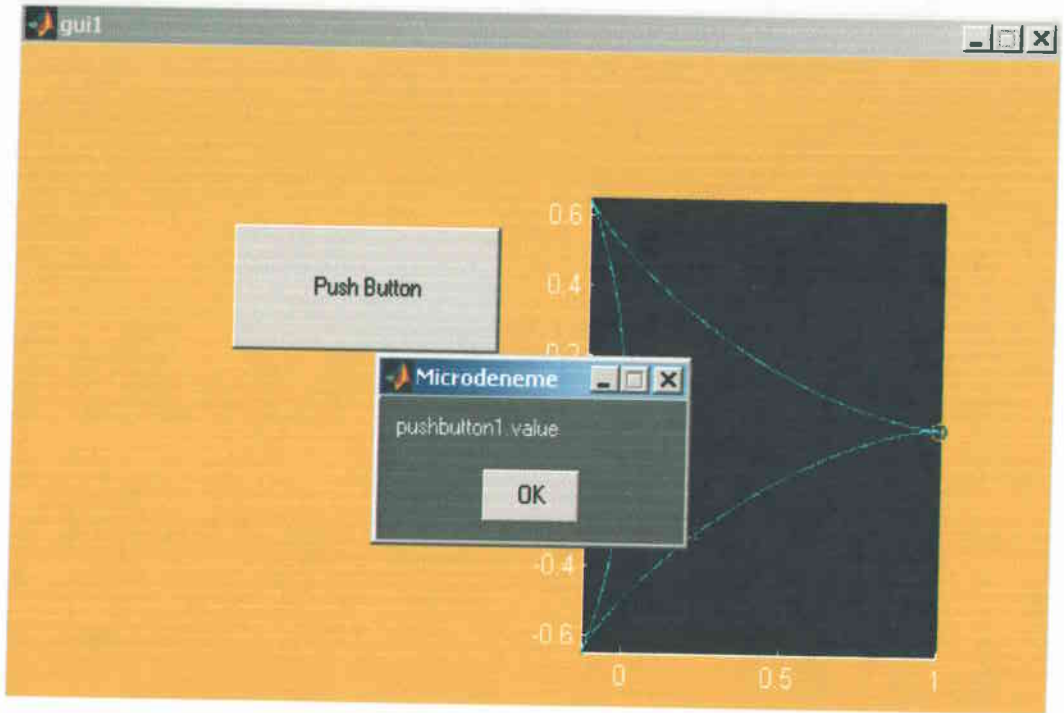
Çalışma alanının arka alan rengini kod ile gerçekleştirmek mümkündür Yukarıdaki temel olayı düşünerek istenilen rengi buraya adapte edebiliriz. Yalnız burada **RGB** fonksiyonu kullanılarak bu işlemler meydana getirilmiştir. **Red – Green – Blue** (RGB) renklerinin kombinasyonlarından oluşan renk bizim rengimiz olacaktır ve 16 777 216 farklı renk üretilebilmektedir. Bu aralıkta istenilen düzende arka alan renkleri seçilebilir.



Şekil 3.6. GUI Düğme oluşturma ve Callback

```
function pushbutton1_Callback(hObject, eventdata, handles)
set(gcf,'Color',[1,0.8,0.3])
% hObject handle to pushbutton1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)
msgbox(pushbutton1.value 'Microdeneme')
whitebg('black')
```

Her bir nesneyi bu şekilde boyutlandırabilir ve arka alan rengini değiştirebiliriz. İşlemler kod ile oluşturulduğundan istenildiğinde grafiksel birim de arayüzün her alanına istenilen grafikde adapte edilebilir.



Şekil 3.7. GUI Düğme oluşturma ve Callback – Hareketli grafik

```

function pushbutton1_Callback(hObject, eventdata, handles)
% hObject   handle to pushbutton1 (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles   structure with handles and user data (see GUIDATA)
t = 0:.01:2*pi;
x = cos(2*t).*(cos(t).^2);
y = sin(2*t).*(sin(t).^2);
comet(x,y);
set(gcf,'Color',[1,0.8,0.3])
msgbox pushbutton1.value Microdeneme
whitebg('black')
t = --10*pi:pi/250:10*pi;
comet3((cos(2*t).^2).*sin(t),(sin(2*t).^2).*cos(t),t);
msgbox('Diğer Yordam')
set(0,'DefaultLineCreateFcn','set(gca,"LineStyleOrder","-.-.-)')

```

sistemi ile yukarıdaki animasyon gerçekleştirilir.

EKSEN Özellikleri (Properties)

Çizelge 3.2. Eksen özellikler

Output Argument	Strings Returned
Mode	'auto' 'manual'
Visibility(Görünürlük)	'on' 'off'
Direction (Yön)	'xy' 'ij'

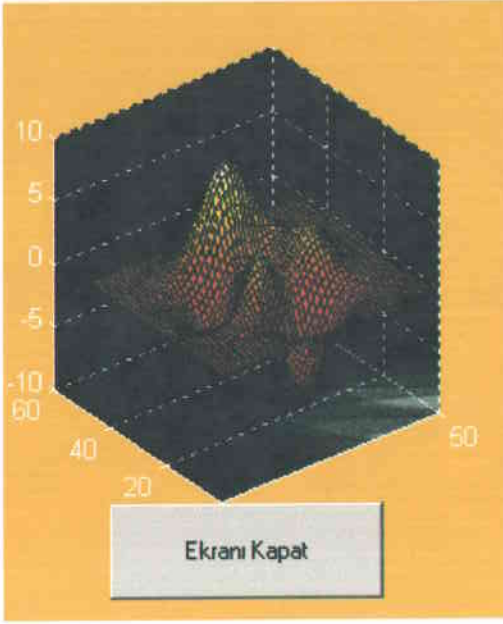
Çizgi Stili

Çizelge 3.3. Çizgi özellikler

Gösterimi	Çizgi Stili
-	solid line (default)
--	dashed line
:	dotted line
-.	dash-dot line


```
set(gcf,'CloseRequestFcn','disp("Bu pencere kapanacaktır..."))
```

Fakat buradaki kod normal düzeyde MATLAB'ı kapatmaya yetmeyecektir. Bunun için ayrı bir kod yazmamamız gerekir.



```
Msgbox ('Ekran Kapanacaktır... ')
```

```
set(0,'ShowHiddenHandles','on')
```

```
delete(get(0,'Children'))
```

Şekil 3.8. CloseRequest ile Figure Sonlandırma

Ekranı çizilen bir grafiğin farklı şekillerde gösterimi için **Uimönü** ile onu tanımlayabiliriz böylece bizim belirleyeceğimiz şekilde ekrana mönü gelecek ve grafik o oranda değişecektir.

```
global i1
```

```
global i2
```

```
global i3
```

```
grafikmönü = uicontextmönü;
```

```
hline = plot(1:10, 'UicontextMönü', grafikmönü);
```

```
cb1 = ['set(hline, "LineStyle", "--)'];
```

```
cb2 = ['set(hline, "LineStyle", ":)'];
```

```
cb3 = ['set(hline, "LineStyle", "-)'];
```

```
i1 = uimönü(grafikmönü, 'Label', 'Aralıklı', 'Callback', cb1);
```

```
i2 = uimönü(grafikmönü, 'Label', 'Noktalı', 'Callback', cb2);  
i3 = uimönü(grafikmönü, 'Label', 'Normal', 'Callback', cb3);
```

Nesnel işleme geçmede bir figür içerisindeki grafiğin limit değerlerinin artışı ve azalışı için aşağıdaki adımlar bir prosedürde tasarlanarak her çağrıldığındaki yerde kullanılmışlardır. Bunun için methodsal bir düzenek aşağıdadır.

```
figure('Position',[245 380 150 90],...  
      'CloseRequestFcn','Aktif',...  
      'Rcsize','off');  
uicontrol('Style','pushbutton',...  
          'Position',[20 20 50 20],...  
          'String','Aşağı',...  
          'Callback',' Mikro-1')  
uicontrol('Style','pushbutton',...  
          'Position',[80 20 50 20],...  
          'String','Yukarı',...  
          'Callback',' Mikro-2')  
uicontrol('Style','edit',...  
          'Position',[45 50 60 20],...  
          'String','1',...  
          'Callback',' Mikro-3');
```



Şekil 3.9. Nested ile Text ve Pushbutton kullanımı

```
function demogui(selector)  
if nargin == 0
```

```

        selector = 0;
    end
    switch selector
    case 0
        fig = figure('Position',[245 380 150 90],...
                    'CloseRequestFcn','demogui(4)',...
                    'Resize','off');
        info.count = 1;
        uicontrol('Style','pushbutton',...
                  'Position',[20 20 50 20],...
                  'String','Aşağı',...
                  'Callback','d(1)')
        uicontrol('Style','pushbutton',...
                  'Position',[80 20 50 20],...
                  'String','Yukarı',...
                  'Callback','d(2)')
        info.editbox = uicontrol('Style','edit',...
                                 'Position',[45 50 60 20],...
                                 'String',num2str(info.count),...
                                 'Callback','d(3)');
        set(fig,'UserData',info,...
            'HandleVisibility','callback')

    case 1
        fig = gcf;
        info = get(fig,'UserData');
        info.count = info.count - 1;
        set(info.editbox,'String',num2str(info.count))
        set(fig,'UserData',info)

    case 2
        fig = gcf;

```

```
info = get(fig,'UserData');
info.count = info.count + 1;
set(info.editbox,'String',num2str(info.count))
set(fig,'UserData',info)
```

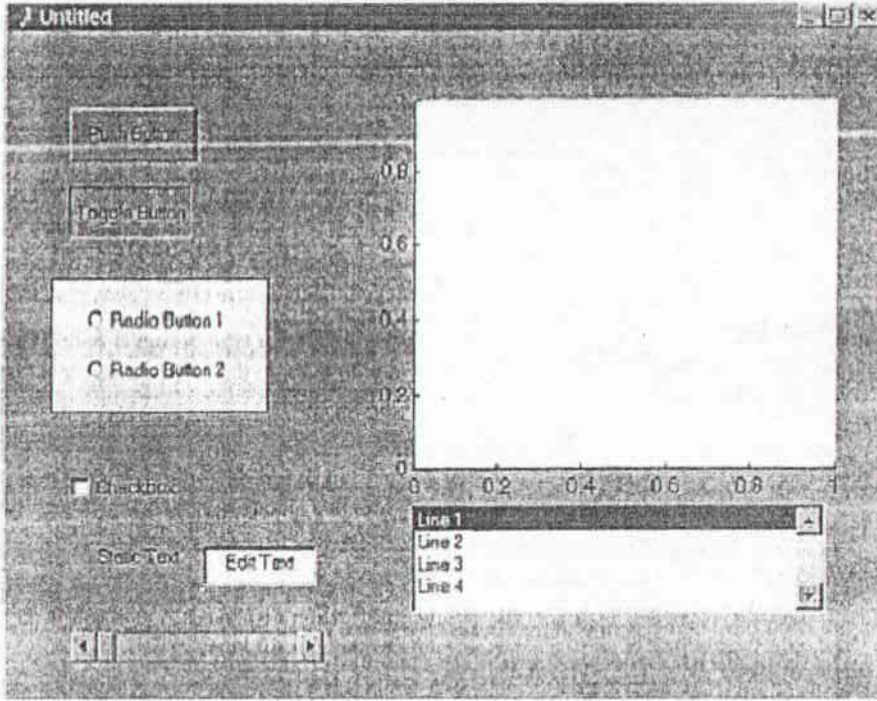
case 3

```
fig = gcf;
info = get(fig,'UserData');
newcount = sscanf(get(info.editbox,'String'),"%d");
if ~isempty(newcount)
    info.count = newcount;
end
set(info.editbox,'String',num2str(info.count))
set(fig,'UserData',info)
```

case 4

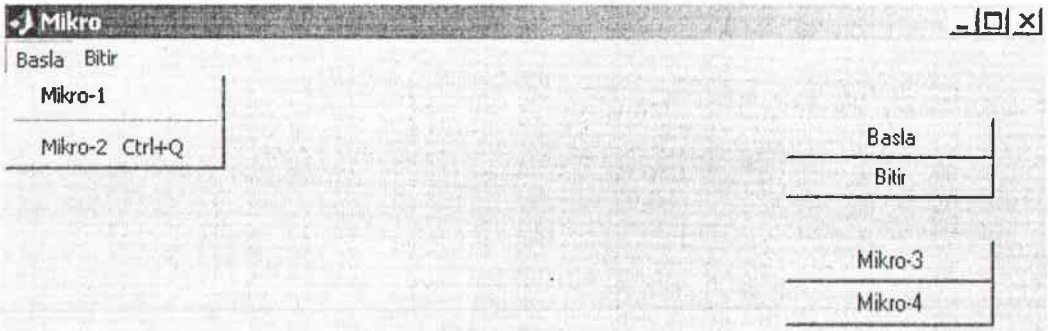
```
cvb=questdlg('Programdan          çıkmak          istediğinize
eminmisiniz?', 'Yes','No','No');
if strcmp(cvb,'Yes')
    delete(gcf)
end
end
```

Yapılacak her iş ayrı bir alt programa adapte edilir. Böylelikle program karmaşık bir yapı yerine daha kullanışlı bir hal almış olacaktır. Bu fonksiyon tanımlamaları yer yer nesnel düzenek veya figure içerisindeki olay yordamlarında bulunmaktadır. Yordamlar içerisinde pushbutton veya edit içerisinde kullanılacak her yapı yukarıdaki sistematığe uygun bir şekilde kodlanacaktır. Ve arayüzlerin esas tasarımına geçilecektir. Esas tasarımı aşağıdaki gibi yapabiliriz..



Şekil 3.10. Nesnelere Tasarım

Mikro adlı figüre penceresini uimönüde tasarlayarak GUI içine nesnelere yerleştirebiliriz. Bunu yaptıktan sonra yukarıdaki ifadeyle birleştirebiliriz.



Şekil 3.11. Nesne ve Mönü Kullanımı

```
clear all;
clc;
hf=figure('NumberTitle','Off','MönüBar','None','Name','Mikro');
hm1=uimönü('Label','Basla');
hm2=uimönü('Label','Bitir');
```

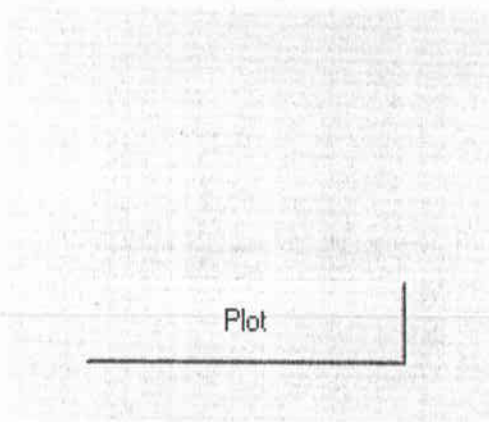
```
hm11=uiönü(hm1,'Label','Mikro-1');
hm12=uiönü(hm1,'Label','Mikro-2',...
'Separator','On',...
'Accelerator','Q',...
'Callback','delete(hf)');
uicontrol('Units','Normalized',...
'Position',[0.75,0.9,0.2,0.05],...
'Style','PushButton',...
'String','Basla',...
'Callback','Deger1=true');
uicontrol('Units','Normalized',...
'Position',[0.75,0.85,0.2,0.05],...
'Style','PushButton',...
'String','Bitir',...
'Callback','Deger2=false');
uicontrol('Units','Normalized',...
'Position',[0.75,0.75,0.2,0.05],...
'Style','PushButton',...
'String','Mikro-3',...
'Callback','delete(gca)');
uicontrol('Units','Normalized',...
'Position',[0.75,0.7,0.2,0.05],...
'Style','PushButton',...
'String','Mikro-4',...
'Callback','delete(gcf)');
```

3.2.2. GUI Örnek Uygulamalar

Burada yukarıdaki ifadeler biraz daha geniş ve basit örneklerle anlatılmıştır. Bu esas programın öğeleri durumundadır ve işlemler bu küçük parçaların bir araya getirilmesi sonucunda oluşacaktır. Bu kısımda daha çok GUI nesnel program oluşumu hakkında bilgiler verilerek örneklendirilmiştir.

3.2.2.1. Örnek uygulama düğme oluşturma

```
hpt = uicontrol(...,'CreateFcn','set(gcf, ...  
"BackgroundColor","blue")')  
x=1:10;  
y=x+1;  
uicontrol('Style','pushbutton', ...  
'String','Plot', ...  
'Callback','plot(x,y)', ...  
'Units','point', ...  
'Position',[50 65 75 35])
```



Şekil 3.12. Pushbutton oluşturma

3.2.2.2. Örnek uygulama dosya çağırma ve disk işlemleri

```
directory_name = uigetdir
```

```
directory_name =
```

```
D:\work\microtol_1
```




Şekil 3.13. Mikrodalga Gözet Penceresi

```
[filename, pathname, filterindex] = uigetfile( ...
```

```
{ '*.mat','MAT-files (*.mat)'; ...
```

```
'*.mdl','Models (*.mdl)'; ...
```

```
'*.*', 'All Files (*.*)'}, ...
```

```
'Pick a file', ...
```

```
'MultiSelect', 'on');
```

```
[filename, pathname, filterindex] = uigetfile( ...
```

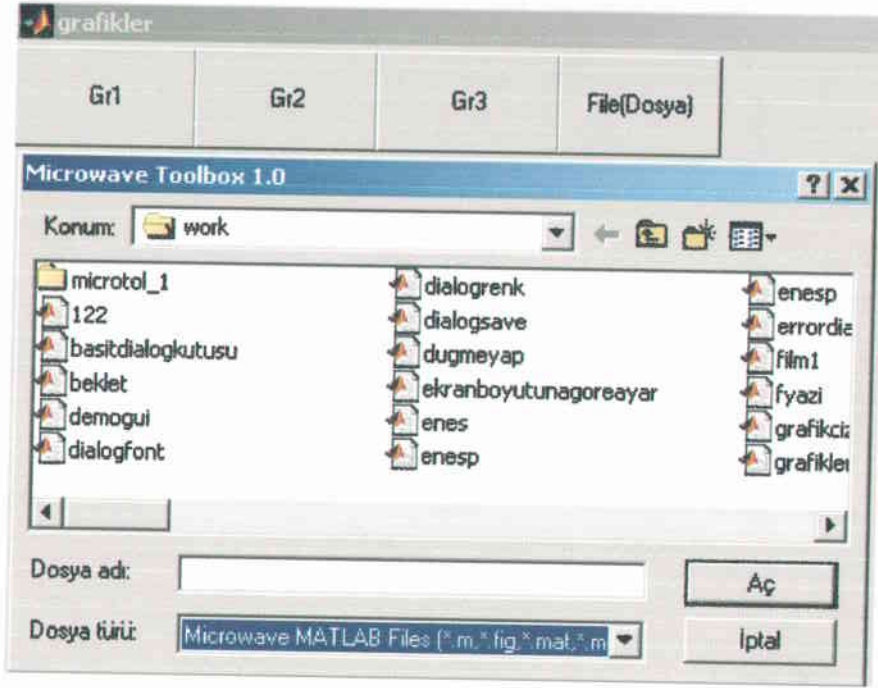
```
{ '*.mat','MAT-files (*.mat)'; ...
```

```
'*.mdl','Models (*.mdl)'; ...
```

```
'*.*', 'All Files (*.*)'}, ...
```

```
'Pick a file', ...
```

```
'MultiSelect', 'on');
```



Şekil 3.14. Mikrodalga Araç Kutusu yol ayarı

Yardım için özel tasarlanmış mönüleride kullanarak istenildiğinde MATLAB dökümantasyon hakkındada bilgi alınabilir.

3.2.2.3. Örnek uygulama – nesnel grup düzeneği

clc

[Dugme1,gosterge1]= uigetpref('graphics',...

'savefigurebeforeclosing',...

'Program Kapanıyor...',...

{'Çıkmazdan kaydetmek istermisiniz?'

'Bu Konu hakkında bilgi almak için yardım seceneğine tıklayınız'

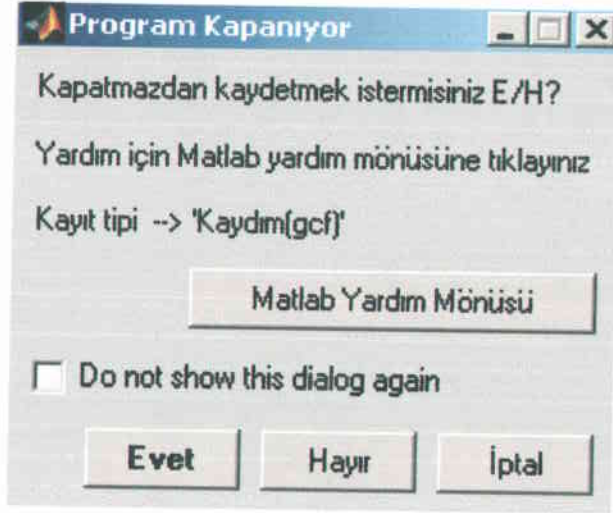
'Kayıt dosya adı','kaydet(gcf)'}},...

{'always','never','Evet','Hayır'}},...

'DefaultButton','Cancel',...

'HelpString','MATLAB Yardım Mönüsü',...

'HelpFcn','doc(''closereq'');')



Şekil 3.15. Seçeneğe göre işlem yapma

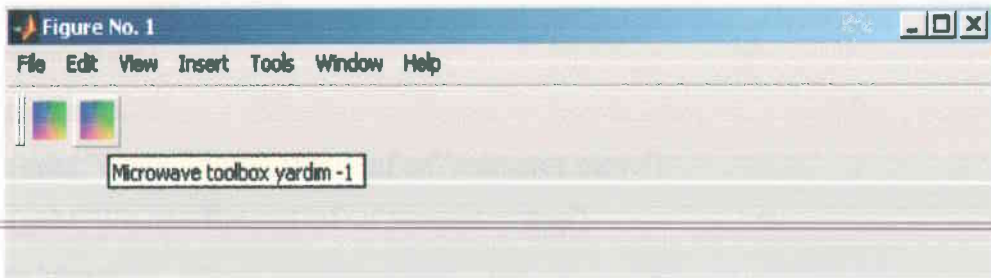
3.2.2.4. Örnek uygulama uipanel ve toolbar hazırlama

```
f = uimönü('Label','Workspace');  
uimönü(f,'Label','New Figure','Callback','figure');  
uimönü(f,'Label','Save','Callback','save');  
uimönü(f,'Label','Quit','Callback','exit',...  
    'Separator','on','Accelerator','Q');  
if strcmp(get(gcbo,'Checked'),'on')  
    set(gcbo,'Checked','off');  
else  
    set(gcbo,'Checked','on');  
end  
set(0,'DefaultUimönüCreateFcn','set(gcbo,...  
    "Visible","on")')
```

```

hpt = uimönü(...,'CreateFcn','set(gcbo,...
"Visible","off")')
h = figure;
hp = uipanel('Title','Main Panel','FontSize',12,...
'BackgroundColor','white',...
'Position',[.25 .1 .67 .67]);
hsp = uipanel('Parent',hp,'Title','Subpanel','FontSize',12,...
'Position',[.4 .1 .5 .5]);
hbsp = uicontrol('Parent',hsp,'String','Push here',...
'Position',[18 18 72 36]);
hpt = uipanel(...,'CreateFcn','set(gcbo,...
"FontName","times","FontSize",14)')
set(uicontrol_handle,'FontName','FixedWidth')
hp = uipanel(...,'Title','\Default');
h = figure('ToolBar','none')
ht = uitoolbar(h)
a = [.05:.05:0.95];
b(:,1) = repmat(a,19,1);
b(:,2) = repmat(a,19,1);
b(:,3) = repmat(flipdim(a,2),19,1);
hpt = uipushtool(ht,'CData',b,'TooltipString','Microwave toolbox yardım ')
hpt1= uipushtool(ht,'CData',b,'TooltipString','Microwave toolbox yardım -1 ')

```



Şekil 3.16. Araç Kutusu Tasarlama

```

        'gui_Callback', []);
    if nargin & isstr(varargin{1})
        gui_State.gui_Callback = str2func(varargin{1});
    end
    if nargin
        [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
    else
        gui_mainfcn(gui_State, varargin{:});
    end
function gui8_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to gui8 (see VARARGIN)
% Choose default command line output for gui8
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);
% UIWAIT makes gui8 wait for user response (see UIRESUME)
% uiwait(handles.figure1);
% --- Outputs from this function are returned to the command line.
function varargout = gui8_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
hObject    handle to pushbutton1 (see GCBO)

```

```

eventdata reserved - to be defined in a future version of MATLAB
handles structure with handles and user data (see GUIDATA)
set(0,'ShowHiddenHandles','yes')
delete(get(0,'Children'))
function pushbutton2_Callback(hObject, eventdata, handles)
hObject handle to pushbutton2 (see GCBO)
eventdata reserved - to be defined in a future version of MATLAB
handles structure with handles and user data (see GUIDATA)
set(handles.text1,'visible','on')
set(handles.text1, 'String', sprintf('Proje --> Doç. Dr. Caner ÖZDEMİR &
Mehmet Sait VURAL'))
Msgbox('Mail gönderme için')
sendmail('saitbm412@yahoo.com','Yeni Başlık', ...
['Mesaj -1' 10 'Mesaj -2' 10 ...
'Mesaj - 4' 10 'Mesaj - 5']);

```

3.2.2.8. Örnek uygulama istenile değere göre işlem



Şekil 3.20. Frekans Ölçüm Animasyonu

```

function gui03(arg)
if nargin == 0
Init;

```

```

else
feval(arg,guihandles(gcbo));
end;
function
Init
fig=figure('MönüBar','none','Name','Gui03','NumberTitle','off','Position',[200,200
,180,150]);
uicontrol('Style','Text','Position',[20,100,60,20],'Tag','Text');
uicontrol('Style','PushButton','String','Fre-100','Position',[20,60,60,20],...
'Callback','gui03 J1','Tag','100');
uicontrol('Style','PushButton','String','Fre-200','Position',[20,20,60,20],...
'Callback','gui03 J2','Tag','200');
uicontrol('Style','ToggleButton','String','Kapalı','Position',[100,60,60,20],...
'Callback','gui03 ToggleButtonPressed','Tag','ToggleButton');
uicontrol('Style','PushButton','String','Kapat','Position',[100,20,60,20],...
'Callback','Close');
function J1(handles)
set(handles.Text,'String','100');
function J2(handles)
set(handles.Text,'String','200');
function ToggleButtonPressed(handles)
if get(handles.ToggleButton,'Value')== 0
set(handles.ToggleButton,'String','Kapalı');
else
set(handles.ToggleButton,'String','Açık');
end;

```

Aynı ifadeleri Radio button ilede yapabildik

```

function RadioButton1(handles)
if get(handles.RadioButton1,'Value')==0
set(handles.RadioButton1,'Value',1);
else

```

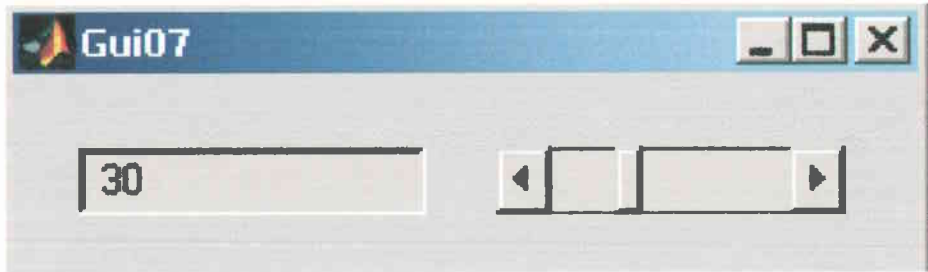


```

set(handles.RadioButton2,'Value',0);
set(handles.RadioButton3,'Value',0);
end;
function RadioButton2(handles)
if get(handles.RadioButton2,'Value')==0
set(handles.RadioButton2,'Value',1);
else
set(handles.RadioButton1,'Value',0);
set(handles.RadioButton3,'Value',0);
end;
function RadioButton3(handles)
if get(handles.RadioButton3,'Value')==0
set(handles.RadioButton3,'Value',1);
else
set(handles.RadioButton1,'Value',0);
set(handles.RadioButton2,'Value',0);
end;

```

3.2.2.9. Örnek uygulama slider (kayan çubuk)



Şekil 3.21. Slider Animasyonu

```

Fig=figure('MönüBar','none','Name','Gui07','NumberTitle','off','Position',[200,200
,260,60]);
uicontrol('Style','Edit','String','50','Position',[20,20,100,20],...
'CallBack','gui07 Edit','HorizontalAlignment','left','Tag','Edit');
uicontrol('Style','Slider','Position',[140,20,100,20],...

```

```
'CallBack','gui07 Slider','Value',50,...  
'Min',0,'Max',100,'Tag','Slider');
```

3.2.3. GUI ile Web Bağlantısı

Yazdığımız programda istenildiğinde bizim hazırladığımız web editorlerine bağlanıldığı gibi Mathworks firmasında bağlanabilecektir. Bu kullanım external file diye ifade edilen programsal yapılar şeklinde alınmış ve GUI yüzeyinin herhangi bir etiketine gönderme yapılar eklenmiştir. Web <http://www.mathtools.net> web mailto:email_address uses your system browser's default e-mail application to send a message to email_address. Web <http://www.mathtools.net> -browser opens a browser to mathtools.net. Then [stat,h1,url]=web returns

```
stat =0
```

```
h1 =com.mathworks.mde.webbrowser.WebBrowser[,0,0,591x140,
```

```
layout=java.awt.BorderLayout,alignmentX=null,alignmentY=null,
```

```
border=,flags=9,maximumSize=,minimumSize=,preferredSize=]
```

```
url =http://www.mathtools.net/
```

```
setCurrentLocation(h1,'http://www.mathworks.com')
```

3.2.4. External Dosya Bağlantısı

External Dosya içerisinde hazır bir GUI özel fonksiyon kullanımı sağlanarak normal GUI'de yapılması çok zor olan işlemler burada daha rahat bir biçimde

kullanılmaktadır. Özellikle nesnelerin sınıf şeklinde tanımlayarak dinamik bir yapı almasında ve Script kullanılmadan normal olarak web bağlantısında bu kullanılmaktadır.

```
% ----- Subfunction PARSECHILDNODES -----  
function children = parseChildNodes(theNode)  
% Recurse over node children.  
children = [];  
if theNode.hasChildNodes  
childNodes = theNode.getChildNodes;  
numChildNodes = childNodes.getLength;  
allocCell = cell(1, numChildNodes);  
children = struct(      ...  
'Name', allocCell, 'Attributes', allocCell, ...  
'Data', allocCell, 'Children', allocCell);  
for count = 1:numChildNodes  
theChild = childNodes.item(count-1);  
children(count) = makeStructFromNode(theChild);  
end  
end  
% ----- Subfunction MAKESTRUCTFROMNODE -----  
function nodeStruct = makeStructFromNode(theNode)  
% Create structure of node info.  
nodeStruct = struct(      ...  
'Name', char(theNode.getNodeName), ...  
'Attributes', parseAttributes(theNode), ...  
'Data', "", ...  
'Children', parseChildNodes(theNode));  
if any(strcmp(methods(theNode), 'getData'))  
nodeStruct.Data = char(theNode.getData);  
else  
nodeStruct.Data = "";
```

```

end
% ----- Subfunction PARSEATTRIBUTES -----
function attributes = parseAttributes(theNode)
% Create attributes structure.
attributes = [];
if theNode.hasAttributes
theAttributes = theNode.getAttributes;
numAttributes = theAttributes.getLength;
allocCell = cell(1, numAttributes);
attributes = struct('Name', allocCell, 'Value', allocCell);
for count = 1:numAttributes
attrib = theAttributes.item(count-1);
attributes(count).Name = char(attrib.getName);
attributes(count).Value = char(attrib.getValue);
end
end
function plot_vars
% Define the GUI layout
h_figure = figure('Units','characters',...
'Position',[72 38 120 35],...
'Color',get(0,'DefaultUicontrolBackgroundColor'),...
'HandleVisibility','callback');
h_axes = axes('Units','characters',...
'Position',[10 4.5 72 26],...
'Parent',h_figure);
h_listbox_label = uicontrol(h_figure,...
'Style','text',...
'Units','characters',...
'Position',[88 29 24 2],...
'String','Select 2 Workspace Variables');
h_listbox = uicontrol(h_figure,...
'Style','listbox',...

```

```

    'Units','characters',...
    'Position',[88 18.5 24 10],...
    'BackgroundColor','white',...
    'Max',10,'Min',1,...
    'Callback',@listbox_callback);
h_popup_label = uicontrol(h_figure,...
    'Style','text',...
    'Units','characters',...
    'Position',[88 13 24 2],...
    'String','Plot Type');
h_popup = uicontrol(h_figure,...
    'Style','popupmönü',...
    'Units','characters',...
    'Position',[88 12 24 2],...
    'BackgroundColor','white',...
    'String',{'Plot','Bar','Stem'});
h_hold_toggle = uicontrol(h_figure,...
    'Style','toggle',...
    'Units','characters',...
    'Position',[88 8 24 2],...
    'String','Hold State',...
    'Callback',{@hold_toggle_callback,h_axes});
h_plot_button = uicontrol(h_figure,...
    'Style','pushbutton',...
    'Units','characters',...
    'Position',[88 3.5 24 2],...
    'String','Create Plot',...
    'Callback',{@plot_button_callback,h_listbox,h_popup,h_axes});
h0 = figure('Units','normalized', ...
    'Color',[0.8 0.8 0.8], ...
    'Name','Linear System Command Center', ...
    'NumberTitle','off', ...

```

```

        'Position',[0.00390625 0.515625 0.96875 0.4166666666666666], ...
        'Tag','Fig1');
h1 = uimönü('Parent',h0, ...
        'Callback','cmdhndlr("mnucmdmsg")', ...
        'Label','&Instructions', ...
        'Tag','mnucmdmsg');
h1 = uicontrol('Parent',h0, ...
        'Units','normalized', ...
        'FontUnits','normalized', ...
        'BackgroundColor',[0.752941 0.752941 0.752941], ...
        'FontSize',0.0547008547008547, ...
        'ListboxTop',0, ...
        'Position',[0.002580645161290323
0.4412903225806452 0.8120000000000001], ...
        'Style','frame', ...
        'Tag','Frame2');
h1 = uicontrol('Parent',h0, ...
        'Units','normalized', ...
        'FontUnits','normalized', ...
        'BackgroundColor',[0.752941176470588
0.752941176470588], ...
        'FontSize',0.0547008547008547, ...
        'ListboxTop',0, ...
        'Position',[0.447334200260078
0.4447334200260078 0.8148148148148148], ...
        'Style','frame', ...
        'Tag','Frame3');
h1 = uicontrol('Parent',h0, ...
        'Units','normalized', ...
        'FontUnits','normalized', ...
        'BackgroundColor',[0 0 0], ...
        'FontSize',0.7619047619047619, ...

```

```

    'ListboxTop',0, ...
    'Position',[0.01300390117035 0.29166666666667 0.42652795838752
0.01851851851852], ...
    'Style','frame', ...
    'Tag','Frame1');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...
    'FontSize',0.32323232323232, ...
    'ListboxTop',0, ...
    'Position',[0.19254032258065 0.8375 0.06754032258065 0.096875], ...
    'Style','frame', ...
    'Tag','Frame2');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...
    'FontSize',0.3440860215053763, ...
    'ListboxTop',0, ...
    'Position',[0.186491935483871 0.44375 0.0735887096774194
0.07187499999999999], ...
    'Style','frame', ...
    'Tag','Frame2');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0 0 0], ...
    'FontSize',0.7111111111111111, ...
    'ListboxTop',0, ...
    'Position',[0.23276983094928 0.67592592592593 0.20156046814044
0.01851851851852], ...

```



```

        'Style','frame', ...
        'Tag','Frame1');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0 0 0], ...
    'FontSize',0.7111111111111111, ...
    'ListboxTop',0, ...
    'Position',[0.67490247074122    0.67592592592593    0.20416124837451
0.01851851851852], ...
    'Style','frame', ...
    'Tag','Frame1');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...
    'FontSize',0.304761904761905, ...
    'ListboxTop',0, ...
    'Position',[0.5967741935483871 0.83125 0.1653225806451613 0.096875],
...
    'Style','frame', ...
    'Tag','Frame2');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0 0 0], ...
    'FontSize',0.7619047619047619, ...
    'ListboxTop',0, ...
    'Position',[0.4577373211963589
0.2916666666666666
0.4278283485045514 0.01851851851851852], ...
    'Style','frame', ...
    'Tag','Frame1');

```

```

h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...
    'FontSize',0.344086021505376, ...
    'ListboxTop',0, ...
    'Position',[0.6401209677419361      0.44375      0.0735887096774194
0.071874999999999999], ...

```

```

    'Style','frame', ...
    'Tag','Frame2');

```

```

h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'Callback','cmdhdlr("btnlocus")', ...
    'FontSize',0.106666666666667, ...
    'ListboxTop',0, ...
    'Position',[0.8971774193548386      0.06875000000000001
0.09879032258064516 0.396875], ...

```

```

    'String','Root Locus', ...
    'Tag','btnlocus');

```

```

h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Callback','cmdhdlr("edtnump")', ...
    'FontSize',0.367816091954023, ...
    'ListboxTop',0, ...
    'Position',[0.3084677419354839      0.70000000000000001

```

```

0.1219758064516129 0.075000000000000001], ...

```

```

    'Style','edit', ...
    'Tag','edtnump');

```

```

h1 = uicontrol('Parent',h0, ...

```

```

'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[1 1 1], ...
'Callback','cmdhdlr("edtdep")', ...
'FontSize',0.3333333333333333, ...
'ListboxTop',0, ...
'Position',[0.3084677419354839
0.1219758064516129 0.07500000000000001], ...
0.60000000000000001
'Style','edit', ...
'Tag','edtdep');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[1 1 1], ...
'Callback','cmdhdlr("edtnumc")', ...
'FontSize',0.367816091954023, ...
'ListboxTop',0, ...
'Position',[0.75 0.696875 0.122983870967742 0.075], ...
'Style','edit', ...
'Tag','edtnumc');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[1 1 1], ...
'Callback','cmdhdlr("edtdenc")', ...
'FontSize',0.3333333333333333, ...
'ListboxTop',0, ...
'Position',[0.75100806451613 0.603125 0.12298387096774 0.075], ...
'Style','edit', ...
'Tag','edtdenc');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...

```

```
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'FontSize',0.952380952380952, ...
'ListboxTop',0, ...
'Position',[0.276129032258065 0.664 0.0296774193548387 0.116], ...
'String','=', ...
'Style','text', ...
'Tag','StaticText2');
```

```
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'FontSize',0.476190476190476, ...
'ListboxTop',0, ...
'Position',[0.236895161290323 0.696875 0.0348387096774194 0.08], ...
'String','N(s)', ...
'Style','text', ...
'Tag','StaticText2');
```

```
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'FontSize',0.476190476190476, ...
'ListboxTop',0, ...
'Position',[0.237903225806452 0.590625 0.0342741935483871 0.08125],
```

...

```
'String','D(s)', ...
```

```
'Style','text', ...
```

```
'Tag','StaticText2');
```

```
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
```

```

'BackgroundColor',[0.752941 0.752941 0.752941], ...
'Callback','cmdhdlr("rdcoefp")', ...
'FontSize',0.4, ...
'ListboxTop',0, ...
'Position',[0.009072580645161289    0.665625    0.163306451612903
0.059375], ...
'String','Enter Coefficients', ...
'Style','radiobutton', ...
'Tag','rdcoefp', ...
'Value',1);
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'Callback','cmdhdlr("rdpzp")', ...
'FontSize',0.4, ...
'ListboxTop',0, ...
'Position',[0.007802340702210663    0.6018518518518519
0.211963589076723 0.06481481481481481], ...
'String','Enter Poles and Zeroes', ...
'Style','radiobutton', ...
'Tag','rdpzp');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.8 0.8 0.8], ...
'FontName','courier new', ...
'FontSize',0.419047619047619, ...
'ListboxTop',0, ...
'Position',[0.01430429128739    0.305555555555556    0.42392717815345
0.10648148148148], ...
'Style','text', ...

```

```

    'Tag','txtnump');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...
    'FontName','courier new', ...
    'FontSize',0.4583333333333333, ...
    'ListboxTop',0, ...
    'Position',[0.01430429128739    0.1944444444444444    0.42392717815345
0.10185185185185], ...
    'Style','text', ...
    'Tag','txtdenp');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0.752941 0.752941 0.752941], ...
    'Callback','cmdhdlr("rdfactp")', ...
    'FontSize',0.4848484848484848, ...
    'ListboxTop',0, ...
    'Position',[0.247074122236671    0.09722222222222221
0.1430429128738622 0.06481481481481481], ...
    'String','Factored Form', ...
    'Style','radiobutton', ...
    'Tag','rdfactp');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'FontUnits','normalized', ...
    'BackgroundColor',[0.752941 0.752941 0.752941], ...
    'Callback','cmdhdlr("rdpolyp")', ...
    'FontSize',0.4637681159420289, ...
    'ListboxTop',0, ...

```

```
'Position',[0.03511053315995 0.10185185185185 0.15994798439532  
0.06481481481481], ...
```

```
'String','Polynomial Form', ...
```

```
'Style','radiobutton', ...
```

```
'Tag','rdpolyp', ...
```

```
'Value',1);
```

```
h1 = uicontrol('Parent',h0, ...
```

```
'Units','normalized', ...
```

```
'FontUnits','normalized', ...
```

```
'BackgroundColor',[0.8 0.8 0.8], ...
```

```
'FontSize',0.41025641025641, ...
```

```
'ListboxTop',0, ...
```

```
'Position',[0.192540322580645 0.45625 0.0604838709677419 0.046875],
```

```
...
```

```
'String','Display', ...
```

```
'Style','text', ...
```

```
'Tag','StaticText1');
```

```
h1 = uicontrol('Parent',h0, ...
```

```
'Units','normalized', ...
```

```
'FontUnits','normalized', ...
```

```
'Callback','cmdhdlr("btnplussp")', ...
```

```
'FontSize',0.5333333333333333, ...
```

```
'ListboxTop',0, ...
```

```
'Position',[0.05548387096774193 0.468 0.08774193548387096 0.044], ...
```

```
'String','+', ...
```

```
'Tag','btnplussp');
```

```
h1 = uicontrol('Parent',h0, ...
```

```
'Units','normalized', ...
```

```
'FontUnits','normalized', ...
```

```
'Callback','cmdhdlr("btnminusp")', ...
```

```
'FontSize',0.6349206349206349, ...
```

```
'ListboxTop',0, ...
```



```

        'Position',[0.2990897269180755                                0.4629629629629629
0.08712613784135241 0.041666666666666666], ...
        'String','- ', ...
        'Tag','btnminusp');
h1 = uicontrol('Parent',h0, ...
        'Units','normalized', ...
        'FontUnits','normalized', ...
        'BackgroundColor',[0.8 0.8 0.8], ...
        'FontSize',0.450128205128205, ...
        'FontWeight','bold', ...
        'ListboxTop',0, ...
        'Position',[0.19556451612903 0.846875 0.06048387096774 0.078125], ...
        'String','Plant', ...
        'Style','text', ...
        'Tag','StaticText1');
h1 = uicontrol('Parent',h0, ...
        'Units','normalized', ...
        'FontUnits','normalized', ...
        'BackgroundColor',[0.8 0.8 0.8], ...
        'FontSize',0.450128205128205, ...
        'FontWeight','bold', ...
        'ListboxTop',0, ...
        'Position',[0.60383064516129                                0.8375                                0.152217741935484
0.084375000000000001], ...
        'String','Compensator', ...
        'Style','text', ...
        'Tag','StaticText1');
h1 = uicontrol('Parent',h0, ...
        'Units','normalized', ...
        'FontUnits','normalized', ...
        'Callback','cmdhdlr("btnminusc")', ...
        'FontSize',0.6349206349206349, ...

```

```

'ListboxTop',0, ...
'Position',[0.7464239271781534                                0.4583333333333333
0.08842652795838751 0.04629629629629629], ...
'String','- ', ...
'Tag','btnminusc');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'Callback','cmdhndlr("btnplusc")', ...
'FontSize',0.5333333333333333, ...
'ListboxTop',0, ...
'Position',[0.5149544863459038                                0.4583333333333333
0.08712613784135241 0.04166666666666666], ...
'String','+', ...
'Tag','btnplusc');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.8 0.8 0.8], ...
'FontSize',0.41025641025641, ...
'ListboxTop',0, ...
'Position',[0.6461693548387096                                0.45625            0.05846774193548387
0.046875], ...
'String','Display', ...
'Style','text', ...
'Tag','StaticText1');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'Callback','cmdhndlr("rdpolyc")', ...
'FontSize',0.4637681159420289, ...

```

```

'ListboxTop',0, ...
'Position',[0.4889466840052    0.10185185185185    0.15864759427828
0.06481481481481], ...
'String','Polynomial Form', ...
'Style','radiobutton', ...
'Tag','rdpolyc');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'Callback','cmdhdlr("rdfact")', ...
'FontSize',0.4848484848484848, ...
'ListboxTop',0, ...
'Position',[0.7009102730819246                                0.1064814814814815
0.1586475942782835 0.06481481481481481], ...
'String','Factored Form', ...
'Style','radiobutton', ...
'Tag','rdfact', ...
'Value',1);
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.8 0.8 0.8], ...
'FontName','courier new', ...
'FontSize',0.4583333333333333, ...
'ListboxTop',0, ...
'Position',[0.45903771131339    0.1944444444444444    0.42522756827048
0.10185185185185], ...
'Style','text', ...
'Tag','txtdenc');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...

```

```

'FontUnits','normalized', ...
'BackgroundColor',[0.8 0.8 0.8], ...
'FontName','courier new', ...
'FontSize',0.4583333333333333, ...
'ListboxTop',0, ...
'Position',[0.4590377113133941    0.3055555555555555    0.4252275682704811
0.1064814814814815], ...
'Style','text', ...
'Tag','txtnumc', ...
'UserData',1);
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'Callback','cmdhndlr("rdpzc")', ...
'FontSize',0.4266666666666667, ...
'ListboxTop',0, ...
'Position',[0.4529032258064516 0.6 0.2129032258064516 0.06800000000000001],
...
'String','Enter Poles and Zeroes', ...
'Style','radiobutton', ...
'Tag','rdpzc');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'Callback','cmdhndlr("rdcoefc")', ...
'FontSize',0.4444444444444444, ...
'ListboxTop',0, ...
'Position',[0.452903225806452 0.668 0.164314516129032 0.059375], ...
'String','Enter Coefficients', ...
'Style','radiobutton', ...

```

```
'Tag','rdcoefc', ...
'Value',1);
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'FontSize',0.476190476190476, ...
'ListboxTop',0, ...
'Position',[0.6794354838709678 0.59375 0.03427419354838709 0.08125], ...
'String','D(s)', ...
'Style','text', ...
'Tag','StaticText2');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'FontSize',0.476190476190476, ...
'ListboxTop',0, ...
'Position',[0.6787096774193548 0.6960000000000001 0.03483870967741935 0.08],
...
'String','N(s)', ...
'Style','text', ...
'Tag','StaticText2');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'BackgroundColor',[0.752941 0.752941 0.752941], ...
'FontSize',0.9523809523809523, ...
'ListboxTop',0, ...
'Position',[0.718709677419355 0.664 0.0296774193548387 0.116], ...
'String','=', ...
'Style','text', ...
```

```

'Tag','StaticText2');
h1 = uicontrol('Parent',h0, ...
'Units','normalized', ...
'FontUnits','normalized', ...
'Callback','cmdlndlr("btnbodedsgn")', ...
'FontSize',0.1035598705501618, ...
'ListboxTop',0, ...
'Position',[0.8951612903225806 0.4875 0.09979838709677419 0.396875], ...
'String','Bode Design', ...
'Tag','btnbodedsgn');
if nargout > 0, fig = h0; end
Hc_frame_1=uicontrol(gcf,'Style','frame',...
'Units','pixels',...
'Position', [25 250 200 230]);
Hc_text_Title=uicontrol(gcf,...
'Style','text',...
'Units','pixels',...
'Position',[25          500          200          12],...
'String',          ['Graphical          User          Interface'],...
'FontSize', [10.0]);
Hc_button_Write_Parameter_File=uicontrol(gcf,...
'Style','pushbutton',...
'Units','pixels',...
'Position',          [25          450          200          15],...
'String',          'Write',...
'Callback', 'Swiggui_Button_Write');
h1 = axes('Parent',h0, ...
'CameraUpVector',[0 1 0],...
'CameraUpVectorMode','manual', ...
'Color',[1 1 1], ...
'GridLineStyle','none', ...

```

```

    'Position',[0.06627101879327399
0.4035608308605341 0.02068965517241379], ...
    'Tag','AxBodePZ', ...
    'XColor',[0 0 0], ...
    'XTickLabelMode','manual', ...
    'YColor',[0 0 0], ...
    'YTickLabelMode','manual', ...
    'ZColor',[0 0 0]);
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...
    'HorizontalAlignment','left', ...
    'ListboxTop',0, ...
    'Position',[0.0712166172106825
0.2106824925816024 0.05517241379310345], ...
    'String','Gain Margin:', ...
    'Style','text', ...
    'Tag','txtgm');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...
    'HorizontalAlignment','left', ...
    'ListboxTop',0, ...
    'Position',[0.5459940652818991
0.2324431256181998 0.05517241379310345], ...
    'String','Gain Crossover @', ...
    'visible','off', ...
    'Style','text', ...
    'Tag','txtgmc');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...

```



```

    'HorizontalAlignment','left', ...
    'ListboxTop',0, ...
    'Position',[0.5509396636993076
0.2176063303659743 0.05517241379310345], ...
    'String','Phase Margin:', ...
    'Style','text', ...
    'Tag','txtpm');
h1 = uicontrol('Parent',h0, ...
    'Units','normalized', ...
    'BackgroundColor',[0.8 0.8 0.8], ...
    'HorizontalAlignment','left', ...
    'ListboxTop',0, ...
    'Position',[0.07220573689416419
0.2245301681503462 0.05517241379310345], ...
    'String','Phase Crossover @', ...
    'visible','off', ...
    'Style','text', ...
    'Tag','txtpmc');

clf
hb = uicontrol('Style','pushbutton')
set(hb, 'String', 'Tarih')
set(hb, 'BackgroundC'olor', [0.2 0.6 1])
c = 0;
set(hb, 'Callback', 'c=c+1, datestr(now)')
set(hb, 'Style', 'checkbox')
set(hb, 'Units', 'pixels', ...
'Position',[100 100 80 25])
set(hb, 'Units', 'normalized', ...
'Position',[0.5 0.5 0.25 0.25])

```

3.3. GÖRSEL PROGRAMLAMA

3.3.1. Visual Basic ile Bağlantı

Visual Basicte yapılan olayda program bize gömülü nesnelere üzerinde OLE nesnesiyle işlemler gerçekleştirilmiştir. Aşağıda verilen ise sadece programın setup için cd yerleştirilerek basit bir animasyonlu işlemler gerçekleştirilmiş şeklidir. Yalnız bu yazılan kodlar sadece tek bir GUI içerisinde tasarlanılan yapıdır. Bu yapı sadece GUI için hazırlanacak yapıların basit bir denemesidir. Bu denemelerden yola çıkılarak bu kodlar her bir fiğüre nesnesi için tasarlanılarak programlanmıştır. Aynı zamanda yine Visual Basicte Flash animasyonlarda eklenmiş ve Visual C++ ile derlenerek MATLAB GUI'de kullanılmıştır. Bu programın görsel olarak bir etkisi ise nesnel bir programla kullanılmasıdır. Nesnel kullanımdaki bazı yapılar,

- Mekanik yapı kontrolü
- Masaüstüne Erişim
- Windows Ortak Dialog Kutularının Çağırılması
- Windows API kullanımları
- Windows API ile Form yüzeyinin değişimleri ile Görsellik
- Windows API ile Çözünürlüklerin ayarlanması
- Windows API ile Sistem hakkında bilgi alma ve işleme göre değiştirme
- Disk ve Dosyalama işlemleri

şeklinde. Yukarıdaki işlemler için deneme.vbp adlı örnek uygulama aşağıda verilmektedir. GUI için tasarlanılan gerçek kodlar programın çalışabilir düzeneği içerisinde.

```
Private Declare Function mciExecute Lib "winmm.dll" (ByVal lpstrCommand  
As String) As Long  
Private Sub Command1_Click()  
mciExecute ("set cd audio door open")  
End Sub
```

```

Private Sub Command2_Click()
mciExecute ("set cdaudio door closed")
End Sub
'Form1'e girecek...
Private Sub Command1_Click()
Dim rc As RECT
Dim r As Long
Dim msg As String
r = SystemParametersInfo(SPI_GETWORKAREA, 0&, rc, 0&)
msg = "Ekran Çözünürlüğü" & vbCrLf
msg = msg & "Genişlik - " & rc.Right & vbCrLf
msg = msg & "Yükseklik - " & rc.Bottom & vbCrLf & vbCrLf
Text1 = msg
Me.Move rc.Left * Screen.TwipsPerPixelX, _
rc.Top * Screen.TwipsPerPixelY, _
rc.Right * Screen.TwipsPerPixelX, _
rc.Bottom * Screen.TwipsPerPixelY
End Sub
Private Sub eklenen_dosya_Click()
Label1.Caption = eklenen_dosya.Caption
End Sub
Private Sub Form_Load()
Label1.Caption = "" 'Son açılan dosyayı temsil eden etiketi temizle
'eklenen_dosya.Visible = False 'Menü seçeneğini gizle
'(Program yeni başladı ve hiç dosya açılmadı)
CommonDialog1.FileName = "*.*)"
End Sub
Private Sub Ac_Click() 'Dosya * Aç komutu çalıştırıldığında
CommonDialog1.DefaultExt = "*.*)" 'Tüm dosyalar
CommonDialog1.Action = 1 'Dosya Aç
eklenen_dosya.Caption = CommonDialog1.FileName 'Genel diyalog kutusunda
seçilen dosyayı

```

```

'Son menü seçeneğine ekle
eklenen_dosya.Visible = True 'Son menü seçeneğini görünür hale getir.
Label1.Caption = CommonDialog1.FileName
End Sub

Declare Function SystemParametersInfo Lib "user32" Alias
"SystemParametersInfoA" (ByVal uAction As Long, ByVal uParam As Long,
ByVal lpvParam As Any, ByVal fuWinIni As Long) As Long
Public Sub GizleCtrlAltDelete(bDisabled As Boolean)
Dim X As Long
X = SystemParametersInfo(97, bDisabled, CStr(1), 0)
End Sub

'option explicit
Dim d As Boolean
Private form_load()
d = True
Module1.Call GizleCtrlAltDelete(d)
End Sub

Private Sub Command1_Click()
MsgBox App.Path
Label1 = "Çalışılan Klasör " & App.Path
End Sub

Private Sub Command10_Click()
Form9.Show
Form1.Hide
End Sub

Private Sub Command11_Click()
Form10.Show
Form1.Hide
End Sub

Private Sub Command13_Click()
Form12.Show
Form1.Hide

```

```
End Sub
Private Sub Command2_Click()
MsgBox App.Path
Label1 = "Çalışılan Klasör " & App.Path
End Sub
Private Sub Command3_Click()
Form2.Show
Form1.Hide
End Sub
Private Sub Command4_Click()
Form3.Show
Form1.Hide
End Sub
Private Sub Command5_Click()
Form4.Show
Form1.Hide
End Sub
Private Sub Command6_Click()
Form5.Show
Form1.Hide
End Sub
Private Sub Command7_Click()
Form6.Show
Form1.Hide
End Sub
Private Sub Command8_Click()
Form7.Show
Form1.Hide
End Sub
Private Sub Command9_Click()
Form8.Show
Form1.Hide
```

```

End Sub
Private Sub Drive1_Change()
On Error GoTo yanlış
ChDrive Drive1.Drive
Dir1.Path = Drive1.Drive
GoTo tamam
yanlış:
MsgBox "Su an bu surucude bilgi yok veya bos"
tamam:
End Sub
Private Sub Dir1_Change()
ChDir Dir1.Path
File1.Path = Dir1.Path
End Sub
Private Sub Form_Load()
Label1 = "Çalışılan Klasör " & App.Path
End Sub
Private Sub Command1_Click()
On Local Error GoTo hata
MkDir Text1
Dir1.Refresh
Exit Sub
hata:
End Sub
Private Sub Dir1_Change()
ChDir Dir1.Path
File1.Path = Dir1.Path
Label1 = "Çalışılan Klasör " & App.Path
End Sub
Private Sub Drive1_Change()
ChDrive Drive1.Drive
Dir1.Path = Drive1.Drive

```

```

End Sub
Private Sub Form_Load()
Label1 = "Çalışılan Klasör " & App.Path
Command1.Caption = "Klasör Oluştur"
Label1.Left = 100
Label1.Top = 100
Drive1.Left = 100
Drive1.Top = Label1.Top + Label1.Height + 20
Dir1.Left = 100
Dir1.Top = Drive1.Top + Drive1.Hcight + 20
Dir1.Height = 2000
File1.Left = 100
File1.Top = Dir1.Top + Dir1.Height + 20
File1.Height = 2000
Me.Height = File1.Top + File1.Height + 500
Me.Width = 9000
Text1.Top = 100
Text1.Left = Label1.Left + Label1.Width + 100
Command1.Top = 100
Command1.Left = Text1.Left + Text1.Width + 100
Text1 = ""
End Sub
Private Sub Command1_Click()
On Local Error GoTo hata
MsgBox File1.FileName
'Secili bir dosya yoksa file1 nesnesinden bosluk dondurur
'Eger secili bir dosya var ise 1.pas verecektir.
d = MsgBox("Silinsinmi", 4, "Onay")
If d = vbYes Then
Kill File1.FileName
Exit Sub
End If

```



```

hata:
End Sub
Private Sub Dir1_Change()
ChDir Dir1.Path
File1.Path = Dir1.Path
Label1 = "Çalışılan Klasör " & App.Path
End Sub
Private Sub Drive1_Change()
ChDrive Drive1.Drive
Dir1.Path = Drive1.Drive
End Sub
Private Sub Form_Load()
Label1 = "Çalışılan Klasör " & App.Path
Command1.Caption = "Dosyayı Sil"
Label1.Left = 100
Label1.Top = 100
Drive1.Left = 100
Drive1.Top = Label1.Top + Label1.Height + 20
Dir1.Left = 100
Dir1.Top = Drive1.Top + Drive1.Height + 20
Dir1.Height = 2000
File1.Left = 100
File1.Top = Dir1.Top + Dir1.Height + 20
File1.Height = 2000
Me.Height = File1.Top + File1.Height + 500
Me.Width = 9000
Command1.Top = 100
Command1.Left = Label1.Left + Label1.Width + 100
End Sub
Private Sub Form_Click()
If CurDir Like App.Path Then
MsgBox "Birbirlerine esittirler"

```

```

Else
MsgBox "esit değildirler"
End If
End Sub
Private Sub Form_Load()
MsgBox "Path  " & App.Path
MsgBox "CompanyName  " & App.CompanyName
MsgBox "EXENAME  " & App.EXENAME
MsgBox "fileDescription  " & App.fileDescription
MsgBox "ProductName  " & App.ProductName
MsgBox "Title  " & App.Title
MsgBox "Curdir " & CurDir
Label1 = CurDir
Label2 = App.Path
End Sub
Private Sub Label2_Click()
End Sub
Private Declare Function CreateEllipticRgn Lib "gdi32" (ByVal X1 As Long,
ByVal Y1 As Long,
ByVal X2 As Long, ByVal Y2 As Long) As Long
Private Declare Function SetWindowRgn Lib "user32" (ByVal hWnd As Long,
ByVal hRgn As Long, _
ByVal bRedraw As Long) As Long
Private Sub Command1_Click()
On Error Resume Next 'Hata görmek istemiyorum
a = 0 Do Until Form1.Height < 300
b = (Form1.Height * Form1.Width) / 1000000
a = a + b
Form1.Top = Form1.Top + a / 2
Form1.Left = Form1.Left + a / 2
Form1.Height = Form1.Height - a
Form1.Width = Form1.Width - a

```

```

Dim hr&, dl&
Dim usew&, usch&
usew& = Me.Width / Screen.TwipsPerPixelX
usch& = Me.Height / Screen.TwipsPerPixelY
hr& = CreateEllipticRgn(0, 0, usew, usch)
dl& = SetWindowRgn(Me.hWnd, hr, True)
DoEvents
Loop
Unload Me 'Form kapansın
End Sub

Private Declare Function fCreateShellLink Lib "STKIT432.DLL" (ByVal _
lpstrFolderName As String, ByVal lpstrLinkName As String, ByVal _
lpstrLinkPath As String, ByVal lpstrLinkArgs As String) As Long
Sub Command1_Click()
Dim lReturn As Long
'Masaüstüne ekle
lReturn = fCreateShellLink("../Desktop", _
"Shortcut to Calculator", "c:\winnt\calc.exe", "")
lReturn = fCreateShellLink("", "Kisayol Hesapmakinesi", _
"c:\windows\calc.exe", "")
'Start up'a ekle.
lReturn = fCreateShellLink("\Startup", "Kisayol Hesapmakinesi ", _
"c:\windows\calc.exe", "")
End Sub

```

3.3.2. Visual Basic Script ve Java Script Kullanımı

Visual Basic Script veya *Java Script* ile dökümanlar web sayfası formatında oluşturulabilmektedir. Daha çok bu işlemler "Yardım" menülerinde veya Mikrodalga için hazırlanmış olan "Demo'lar" daha çok kullanılacaktır. Mesela program çalıştığında ekrana bir liste kutusunu web sayfası formatında kullanıcıya sunup, kullanıcının buradan seçeceği seçeneğe göre programın "Türkçe" veya "İngilizce"

seçeneklerinden hangisinin kullanılacağı sorusuna, yapılan tezde görev alan danışman ve tezi yapan kişi hakkındaki bilgilerin bir power point slaytı (.PPS formatında) şeklinde ekrana gelmesi ve sırayla programın ana ekran görüntülerinin bu web sayfasında görüntülenmesi olayını da bu kodlarla yapılmıştır. Bu listenin oluşum örnek kodu aşağıda verilmiştir.

```
function ListBoxStyle () {
    var style = new Style(); //
    style.styleName='ListBoxStyle';
    style.mOverColor = style.selBackColor;
    style.firstRowColor = '#FFFFFF';
    style.altRowColor = '#FFFFFF';
    style.loadImages = function() {
        if(!this.imageOn)this.imageOn =
        Styles.getImage('check_on.gif',13,13);
        if(!this.imageOff) this.imageOff =
        Styles.getImage('check_off.gif',13,13);
    };

    style.initStyle = function() {
        this.vscBar.setAnchor({top:1,right:1,bottom:1});
        this.renderStyle();
    };
    style.renderStyle = function(act) {
        var all=!act;

        if(all){
            this.lyrItms.setBgColor(this.getStyleAttribute('firstRowColor'));
            this.setBgColor(this.getStyleAttribute('backColor'));
            if(this._created) this._modItemsLayout();
        }
    };

    return style;
}
```

```
}
```

3.3.3. Xml Formatında Veri Oluşturma

Bilgilerin *XML* ile ifade edilmesi sağlanarak kullanım yerine göre bu sistemde faydalanılmıştır.

```
% Create a sample XML document.  
docNode = com.mathworks.xml.XMLUtils.createDocument...  
('root_element')  
docRootNode = docNode.getDocumentElement;  
for i=1:20  
    thisElement = docNode.createElement('child_node');  
    thisElement.appendChild...  
        (docNode.createTextNode(sprintf('%i',i)));  
    docRootNode.appendChild(thisElement);  
end  
docNode.appendChild(docNode.createComment('Sayfa -1 Mikrokayit'));  
% Save the sample XML document.  
xmlFileName = [tempname,'.xml'];  
xmlwrite(xmlFileName,docNode);  
edit(xmlFileName);
```

3.4. DERLEME

Herhangi bir programla dili ile hazırlanan bir uygulamanın, hazırlanan platformdan (MATLAB, Visual Basic, Visual C) bağımsız olarak çalıştırılabilmesi için derleme (compile) yapılır. Hazırlanmış platformdan bağımsız olarak çalışan bu uygulamalar, yapıldıkları bu programlara ihtiyaç duymadan herhangi bir bilgisayarda çalışabilirler.

Derleme, bir uygulamanın kullanılan işletim sistemine göre çalıştırılabilir (executable) programlar oluşturulması anlamına gelmektedir. MS Windows 95, MS Windows 98, MS Windows 2000, MS Windows XP, MS Windows NT 4.x, MS Windows NT 5.x gibi işletim sistemlerinde çalıştırılabilir dosyalar EXE uzantılı MS-DOS veya Win32 (GUI) uygulamalarıdır.

MATLAB'da hazırlanan M-fonksiyon'lar ve GUI uygulamaları (.fig uzantılı dosyalar) derlenebilir. Böylelikle MS Windows'da derlenen bir MATLAB uygulaması, MS Windows ailesine ait tüm işletim sistemlerinde Win32 tabanlı EXE uzantılı dosyalar olarak çalışabilirler.

Matlab'da derleme için "*MathWorks*" firması tarafından dağıtılan *MATLAB Compiler*'a ihtiyaç vardır. MATLAB Compiler, hazırlanan M-fonksiyon dosyalarına, seçilen bir derleyiciye göre derleme işlemi yapar. Yani bir M-fonksiyon, Borland C++, Borland C++ Builder, Compaq Visual Fortran, Digital Visual Fortran, ve Microsoft Visual C/C++ dillerine göre derleneceği gibi, MATLAB compiler ile gelen Matlab derleme aracı olan "Lcc" ile de derlenebilir.

MATLAB compiler ancak başlangıç ibaresi olarak fonksiyon belirteci olan dosyaları derlerler. Buradanda anlaşılacağı üzere M-dosyaları derleyemezler. Bundan dolayı MATLAB'da M-fonksiyon hazırlamaya yatkınlık son derece önemlidir.

MATLAB derleme işleminde en çok kullandığımız komutlar MCC ve MBUILD'dir. MCC, MATLAB programından C veya C++ programına çevirirken kullanılan komutumuzdur. Bu uygulamalar önce C/C++'a çevrilir bundan sonrada bu çevrim işletim sistemine uygun çalıştırılabilir uygulamalara dönüştürülür.

MBUILD komutu ise, hazırda bulunan C ve FORTRAN komutlarını derlemede kullanılır. MCC komutu MBUILD komutunu kendi içerisinde barındırma özelliğindedir. Esasen MATLAB compiler ilk önce MATLAB uygulamalarını MCC ile C/C++ kodlarına, ardından bu kodları MBUILD ile çalıştırılabilir dosyalara çevirir. Genel kullanımı :

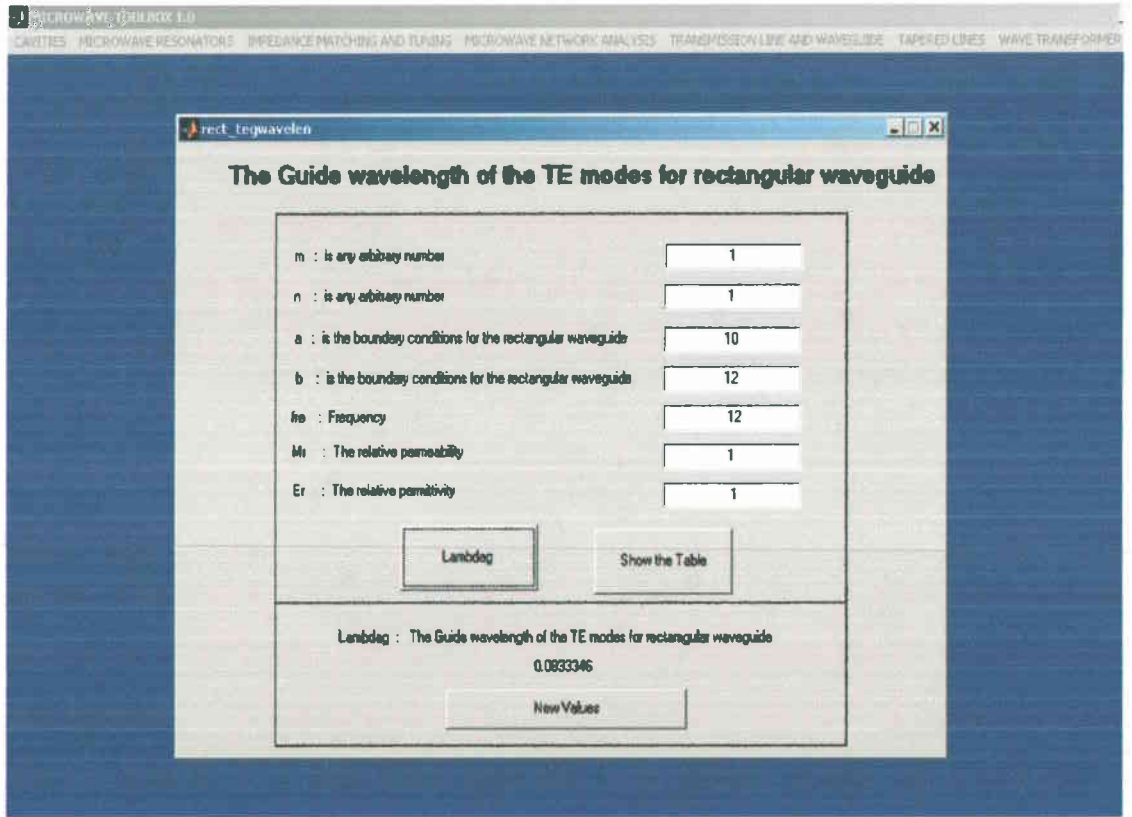
Mcc – seçenek dosya adı şeklindedir. Girilen dosya adı derlenecek fonksiyonu belirtirken seçenek ise, derleme sırasında bir hata olmaması için hazırlanan uygulamanın yapısına göre bir tlr seçimidir.

Çizelge 3.5. Mcc compiler seçenekleri

Seçenek	Açıklama
-m	C tabanlı uygulama oluşturur.
-p	C++ tabanlı uygulama oluşturur.
-B sgl	C tabanlı grafik kütüphaneli uygulama oluşturur ve grafik arabirimli MATLAB uygulamaları için kullanılır.
-B sglepp	C++ tabanlı grafik kütüphaneli uygulama oluşturur ve grafik arabirimli MATLAB uygulamaları için kullanılır.

Bu seçeneklerden en önemlisi -m ve -B sgl'dir M-foksiyon olarak hazırlanan dosyalar - m seçeneği ile GUI olarak hazırlanalar ise -B sgl ile daha çok derlenmektedirler.

karşımıza TM ile hazırlanmış program çıkacaktır. Programın özeti bu noktada belli olmaktadır. Çünkü kullanıcı bu konuya hakim olmadan bile Şekil 4.3.’teki simülasyon programını görsel bir platformda kullanabilmektedir. Hesaplanacak olan “TE Modundaki Dikdörtgensel dalga klavuzunun dalga uzunluğu” probleminin hesabını programcıya bırakarak “Labdag” adlı düğmeye tıklayarak neticeyi bir sonuç kutusu içerisinde görebilecektir.



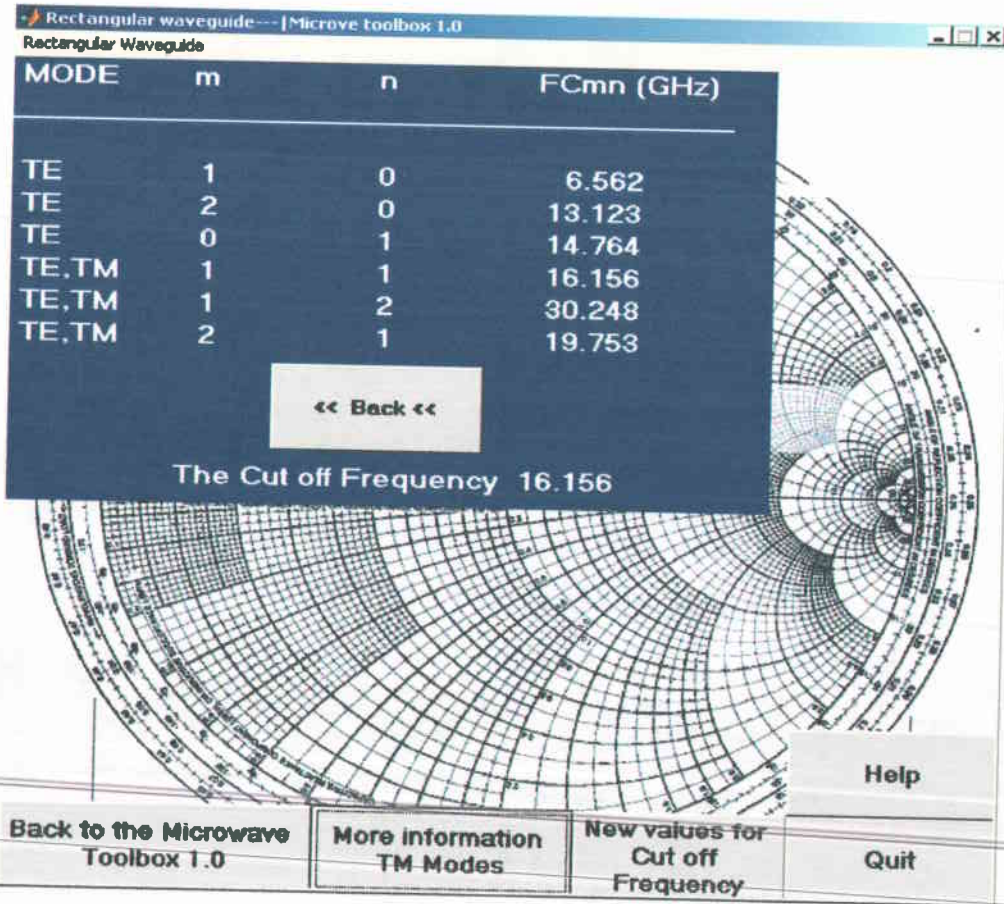
Şekil 4.3. Dikdörtgensel dalga uzunluğu simülasyonu

Uygun bir örnek görmek istediğindede “Show the Table” düğmesine tıklayarak o konu ile ilgili tabloya erişebilecek ve istenildiğinde ana programa tekrar geçebilecektir.

Çizelge 4.1. Dikdörtgensel dalga kılavuzunda TE ve TM modları için m ve n değerlerine göre kesim frekans değerleri

Mode	m	n	$f_{c_{mn}}$ (GHz)
TE	1	0	6.562
TE	2	0	13.123
TE	0	1	14.764
TE,TM	1	1	16.156
TE,TM	1	2	30.248
TE,TM	2	1	19.753

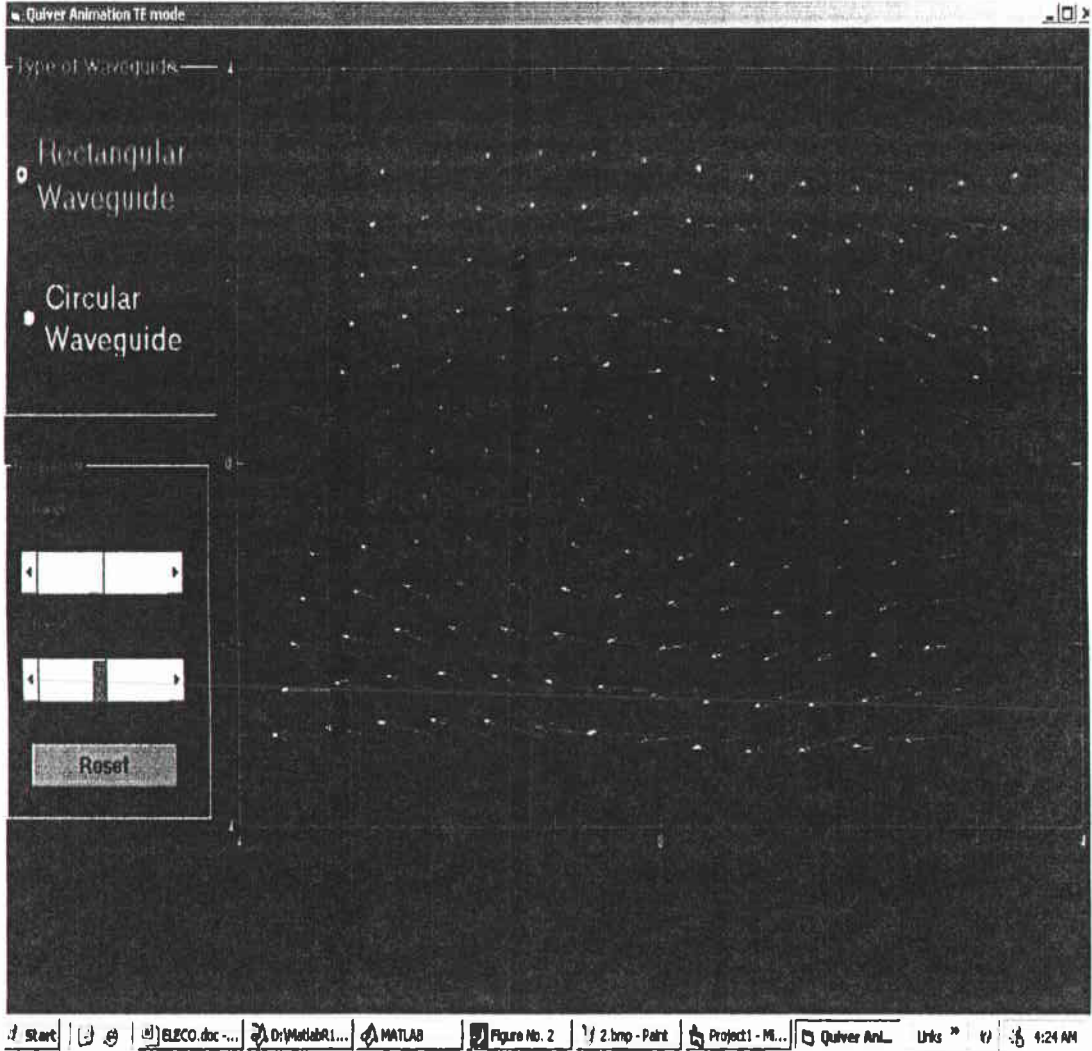
Çizelge 4.1.' de çıkması gereken sonuçları Şekil 4.5.'teki simülasyon programını sonucunda da elde edebiliriz.



Şekil 4.5. Dalga kılavuzu için araç kutusunun hesapladığı kesim frekans değerleri

Mikrodalga araç kutusu çalıştırıldığında, Şekil 4.5'de görülen kesim frekans yanıtları alınacaktır.

Ayrıca, dalga klavuzunun içinde oluşan alan değerlerinin animasyonunu da elde etmek mümkündür (Şekil 4.6).



Şekil 4.6. Değerleri girilen dalga kılavuzu için sunulan alan animasyon

İkinci örnek olarak, bir dairesel dalga kılavuzunun dalga empedans benzetimi gösterilmiştir. Dairesel dalga kılavuzu için alan bileşenleri aşağıdaki gibidir [9]:

```

function [Rnm]=circ_freq(a,d,l,fre,Nr,Er)
sec=input('TE or TM ');
if sec=='TE'
    if nargin==4
        clc
        R=[ 3.832 7.016 10.174 ; 1.041 5.331 8.536; 3.054 6.706 9.970];
        disp(' Rnm values ');
        disp('-----');
        for i=1:3
            for j=1:3
                fprintf(' %f\n',R(i,j))
            end
        end
        pause
        clc
        n=input('n : ');
        m=input('m : ');
        test1 = ((n>=0) & (n<=2));
        test2 = ((m>=1) & (m<=3));
        n=n+1;
        if (test1 & test2)
            Rnm=R(n,m);
            kc=(Rnm)/a;
            fprintf('The Cut off wavenumber %f\n',kc);
        else
            error('You must control m and n values ', 'Warning', 'modal')
        end
        w=2*pi*fre;
        k=w*sqrt(Nr*Er)
        Freq=(c)/(2*pi*sqrt(Nr*Er))*sqrt(kc+(1*pi/d)^2);
        fprintf('The Resonant frequency %f\n',Freq);
    elseif (nargin==3) | (nargin==2) | (nargin==1)
        error('You must enter (a,fre,Nr,Er) parameters ', 'Warning', 'modal')
    else
        error('You must enter parameters', 'Warning', 'modal')
    end
end

```



```

else
    if nargin==4
       clc
        R=[ 2.405 5.520 8.654; 3.832 7.016 10.174; 5.135 8.417 11.620];
        disp(' Rm values ');
        disp('-----');
        for i=1:3
            for j=1:3
                fprintf(' %f\n',R(i,j))
            end
        end
        pause
        clc
        n=input('n : ');
        m=input('m : ');
        test1 = ((m>=0) & (m<=2));
        test2 = ((m>=1) & (m<=3));
        n=n+1;
        if (test1 & test2)
            Rm=R(n,m);
            kc=(Rm)/a;
            fprintf('The Cut off wavenumber %f\n',kc);
        else
            errorDlg('You must control m and n values ', 'Warning', 'modal')
        end
        w=2*pi*fre;
        k=w*sqrt(Mr*Er);
        Freq=(c)/(2*pi*sqrt(Mr*Er))*sqrt(kc+(1*pi/d)^2);
        fprintf('The Resonant frequency %f\n',Freq);
        errorDlg('You must enter (a, fre, Mr, Er) parameters ', 'Warning', 'modal')
    else
        errorDlg('You must enter parameters', 'Warning', 'modal')
    end
end
end

```

Yukarıda hesaplanan m-dosyasında

```

al=findobj(gcf,'Tag','edit1');
a=get(al,'String')
a=str2num(a);

```

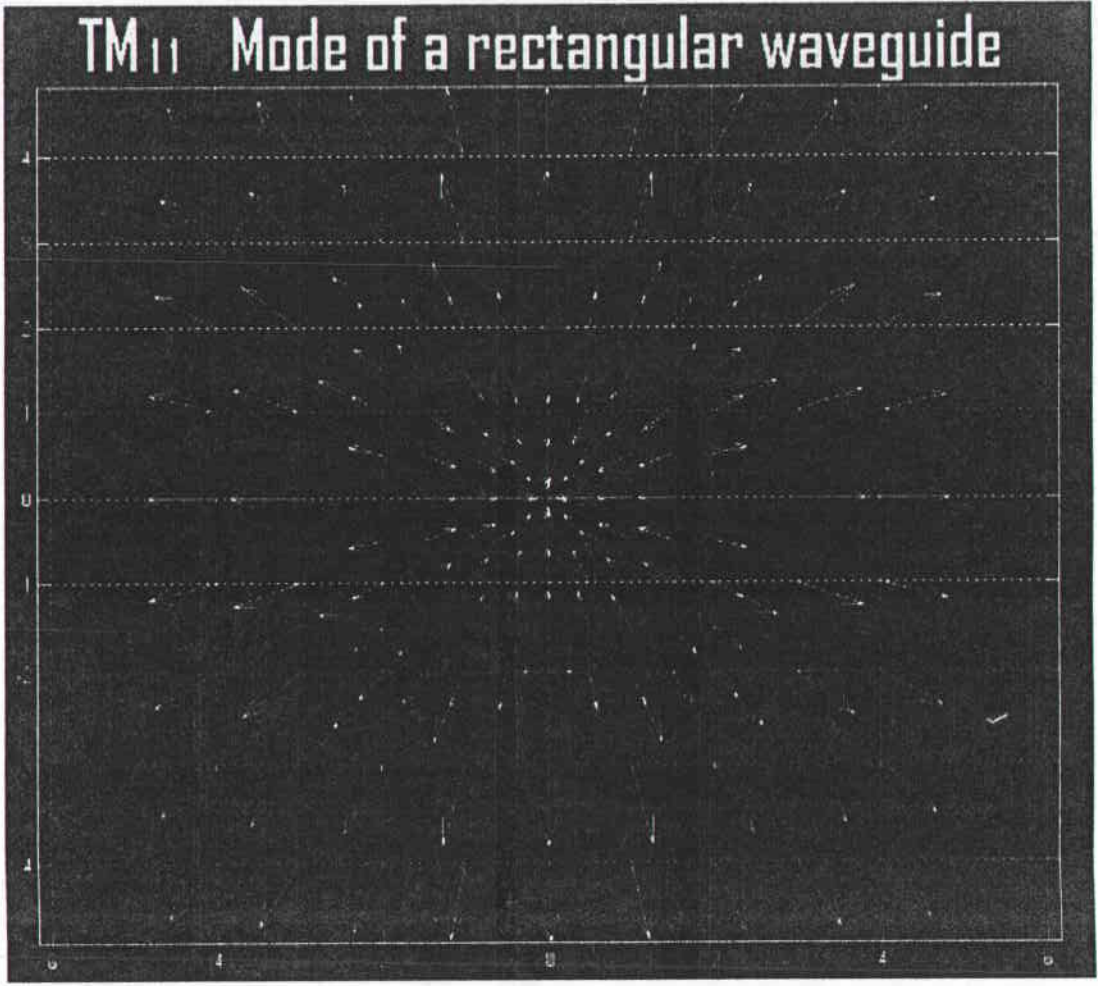
işlemi *fre*, *Mr*, *Er* değerleri için de yapılmaktadır ve 'edit' kutularından alınan bu değerler kontrol edildikten sonra yukarıdaki m-dosyasındaki gibi hesaplanarak,

```
Sonuc=findobj(gcf,'Tag','Yaz')  
Set(str3,'String',Zte)
```

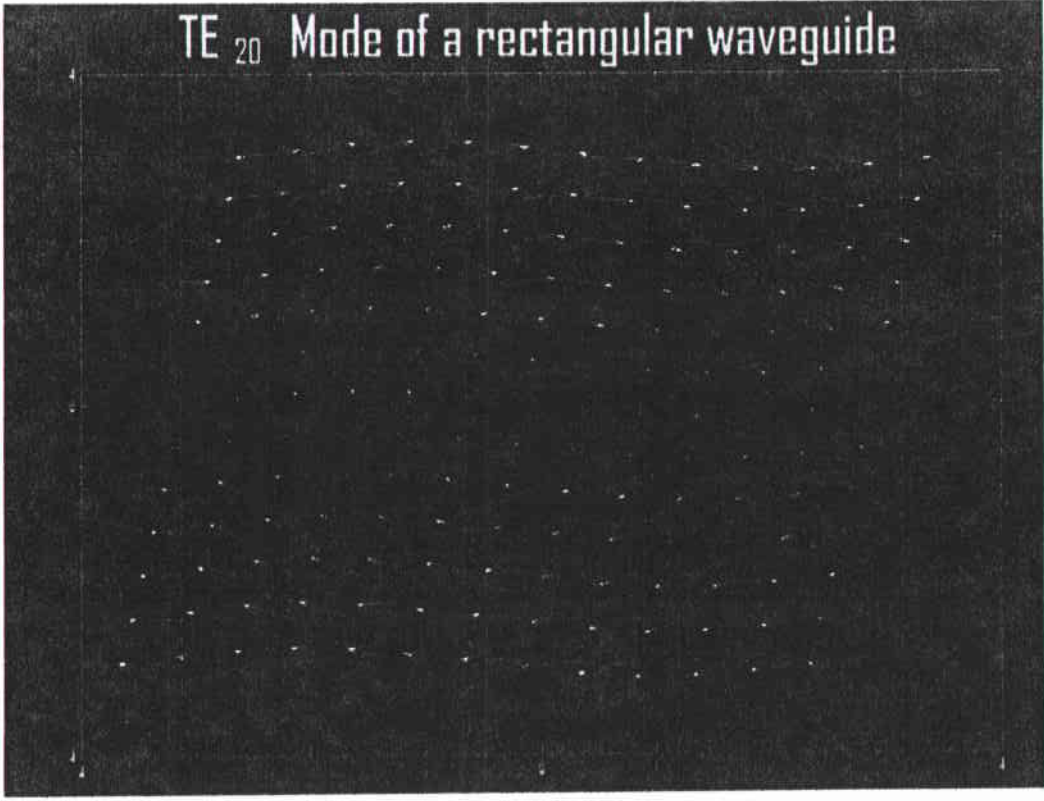
program parçası ile "yaz" adlı statik metin kutusuna yazdırılmış olur. Böylece, bir m - dosyası görsel yapı içerisine aktarıldığında "nargin" giriş, "nargout" çıkış kontrol elemanları kullanılmadan, metin kutuları, etiketler ile parametre değerleri elde edilerek, gerekli hesaplamalar yapıp tekrar görsel platforma sonuçlar aktarılmış olmaktadır.

Böylece kullanıcının yanlış bir parametre girmesi veya yanlış işlem yapması önlenmiş olur.

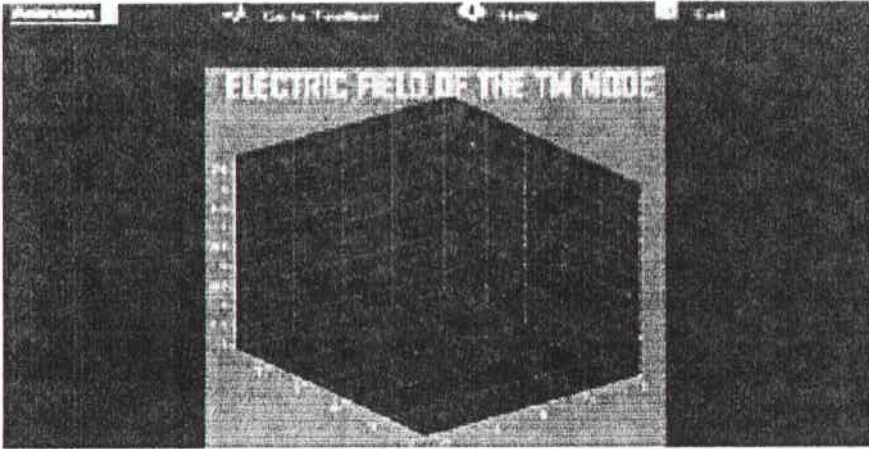
Hazırlanan mikrodalga araç kutusuna, Elektrik ve Manyetik alan animasyon özelliği de eklenmiştir. Şekil 4.7.'de bir dikdörtgensel dalga klavuzu için klavuzunun içindeki alan çizgilerinin animasyonları görülmektedir.



Şekil 4.7. Dairesel dalga klavuzunda TM₁₁ modu elektrik alan animasyonu.



Şekil 4.8. Dikdörtgensel dalga klavuzunda TE20 elektrik alan animasyonu.



Şekil 4.9. Dikdörtgensel dalga klavuzunda TM11 3 boyutlu animasyonu.

sadece gerekli parametreleri girerek, görsel olarak mikrodalga uygulamaları yapmak mümkün olmaktadır. Ayrıca hazırlanan yardım mönüsü ile o konu hakkında teknik bilgi, teoremin açıklamaları, kullanılan değişkenlerin giriş ve çıkış olarak birimleri, m-dosyasının içeriği gibi son kullanıcıya kolaylıklar sağlayan bilgiler sunulmaktadır.

Yazılım Mikrodalga Mühendisliği için yararlı olabilecek bir program hazırlayarak sadece bir master tezi olarak değil üniversitelerin Elektrik-Elektronik Mühendisliği bölümlerinde kullanılabilir bir simülasyon programı özelliği de katılmıştır. Zaten bu tez çalışmasında MATLAB programının özelliklerini diğer nesnel programlarla birleştirilmesi hedeflenmiş ve başarılmıştır. Böylece ortak uygulanabilir bir program oluşturup bu program içerisinde de *Mikrodalga* teoremleri kullanılması sağlanmıştır. Bir diğer nokta ise kullanıcı açısından, tez kullanıcının en az hata ile görsel bir platformda, yardım mönüleriyle ve diğer işlemlerle Mikrodalga Mühendisliğine faydalı olabilecek bir yapı kazandırılması olmaktadır.

5.2. ÖNERİLER

Bu çalışmadan sonra MATLAB'ın diğer araç kutularında tasarlanabilir. Hatta bu tezde sıkça kullanılan .COM ve nesnel dillerden oluşan bir fonksiyonel program yapısı oluşturularak MATLAB'ın içerisine entegre edilirse, bir *.Dll* kütüphanesi kullanıyormuşuz gibi diğer kullanıcılara GUI ile oluşturacakları kolay programlar sunulabilir. Bu yolla MATLAB programı bir çok araç kutusundan oluşan bir "*Visual MATLAB*" (Görsel MATLAB) haline getirilebilir. Zaten ToyotaSA kendi fabrikalarındaki kullanıma uygun buna benzer bir sistem geliştirmektedir.

Tez çalışmasında kullanılan yazılım, görsel programların MATLAB içerisinde kullanılabilirliğini göstermiştir. İster Visual Basic ister Visual C++ ister bir başka dil olsun bu dillerin de MATLAB'da yerine göre kullanımları sağlanmalıdır. MATLAB programcısı tarafından hazırlanabilecek bir yazılım ile MATLAB'a görsel bir arayüz ekleyerek görsel dillerin kullanımı sağlanabilir. Aynı zamanda bunun terside söz konusudur. MATLAB'da integral hesaplamak için tek bir komut yeterli olurken diğer programlama dillerinde bunu yapmak o kadar kolay olmamaktadır. Hesaplamanın bu yapı içerisinde sağlanması için integrali var olan bir kaç yöntem ile birlikte kendi algoritmik ifadelerimizle oluşturmak çok fazla zamanımızı alan bir

KAYNAKLAR

- [1] Deschamps, G. A. , "Microstrip Microwave Antennas", 3rd USAF Symposium on Antennas, (1953).
- [2] Gutton, H., and G. Baissinot, "Flat Aerial for Ultra High Frequencies", French Patent No. 70313, (1955).
- [3] Pozar, D. M., "Microwave Engineering", Univ. of Mass. at Amherst, John Wiley & Sons, New York Inc. ISBN: 0-471-17096-8; pp. 715, (1997).
- [4] Hanselman, D. and Littlefield, B., "Mastering Matlab", Prentice –Hall Int, Upper Saddle River, New Jersey, Inc. ISBN: 0-13-858366-8, pp. 638, (1998).
- [5] Biran, A. and Breiner, M., "Matlab For Engineers", Addison-Wesley Pub. Co., pp. 535, (1995).
- [6] Uzunoğlu, M., Kızıllı, A. ve Onar, Ö. Ç., "Matlab 6.0 – 6.5", Türkmen Kitabevi, İstanbul, 517 (s), (2002).
- [7] Güneş, A. ve Yıldız, K., "Matematik ve Grafik Programlama Dili", Türkmen Kitabevi, İstanbul, 212 (s), (1997).
- [8] Oltman, H. G. and Huebner, D. A., "Electromagnetically Coupled Microstrip Dipoles", IEEE Transactions on Antennas and Propagation, Vol. **29**, No. 1, pp. 151 - 157, (January 1981).
- [9] Collin, R. E., "Foundations for Microwave Engineering", McGraw-Hill Book Co., New York, pp. 335, (1992).
- [10] Harrington, R. F., "Time Harmonic Electromagnetic Fields", McGraw Hill, New York, pp. 432, (1961).
- [11] Choi, Y. J., McCarthy, K. L., McCarthy, M. J. and Mosig J. R., "A MATLAB graphical user interface program for tomographic viscometer data processing, Static Green's functions with Conformal Mapping and MATLAB", IEEE Antennas and Propag. Mag., Vol. **45**, No.5, pp. 123-135, (Oct.2003).
- [12] Echols, M. M. , Smith D. K. and , Nirschl, D. S., "A Web-Based Instrument Monitoring System" , Journal of the Association for Laboratory Automation, Vol. **9**, No.6, pp. 398-403, (December.2004).
- [13] Baskır, I. M. and Drozd, A. V., "New Matlab software for wavelength selection", Chemometrics and Intelligent Laboratory Systems, Vol. **6**, pp. 89-

91, (20 December 2002).

- [14] Prakotpol, D. and Srinophakun, T., "Gapinch: Genetic Algorithm Toolbox For Water Pinch Technology", King Mongkut's University of Technology Thonburi, Bangkok 10140, Vol.;43, pp. 203–217, (2004).
- [15] Pozar, D. M., "Rigorous Closed - Form Expressions for the Surface Wave Loss of the Printed Antennas", *Electronic Letters*, Vol. 26, pp. 954 -956, (1990).
- [16] James, J. R. and Henderson A., "High-Frequency Microstrip Open - Circuit Terminations," *IEEE J. Microwaves, Optics and Acoustics*, Vol. 3, pp. 205 - 218, (1979).
- [17] Balanis, C. A., "Advanced Engineering Electromagnetics", John Wiley&Sons, New York, pp. 238, (1989).
- [18] Aberle, J. T. and Zavosh F., "Analysis of Probe - Fed Circular Microstrip Patches Backed by Circular Cavities", *Electromagnetics*, Vol. 14, pp. 239-258, (1994).
- [19] Nowicky, T. E., "Microwave Substrates Present and Future", *Proc. Workshop on Printed Circuit Antenna Technology*, New Mexico State University, Las Cruces, NM, Vol. 5, pp. 26. 1 - 26. 12, (Oct.1979).
- [20] Alexopoulos, N. G., "Integrated - Circuit Structures on Anisotropic Substrates," *IEEE Trans. Microwave Theory and Techniques*, Vol. 33, pp. 847 - 881, (1985).
- [21] Anonim, "Matlab Guide Kullanımı". Mathworks Homepage, Erişim, <http://www.mathworks.com>, [20 Mayıs 2005].