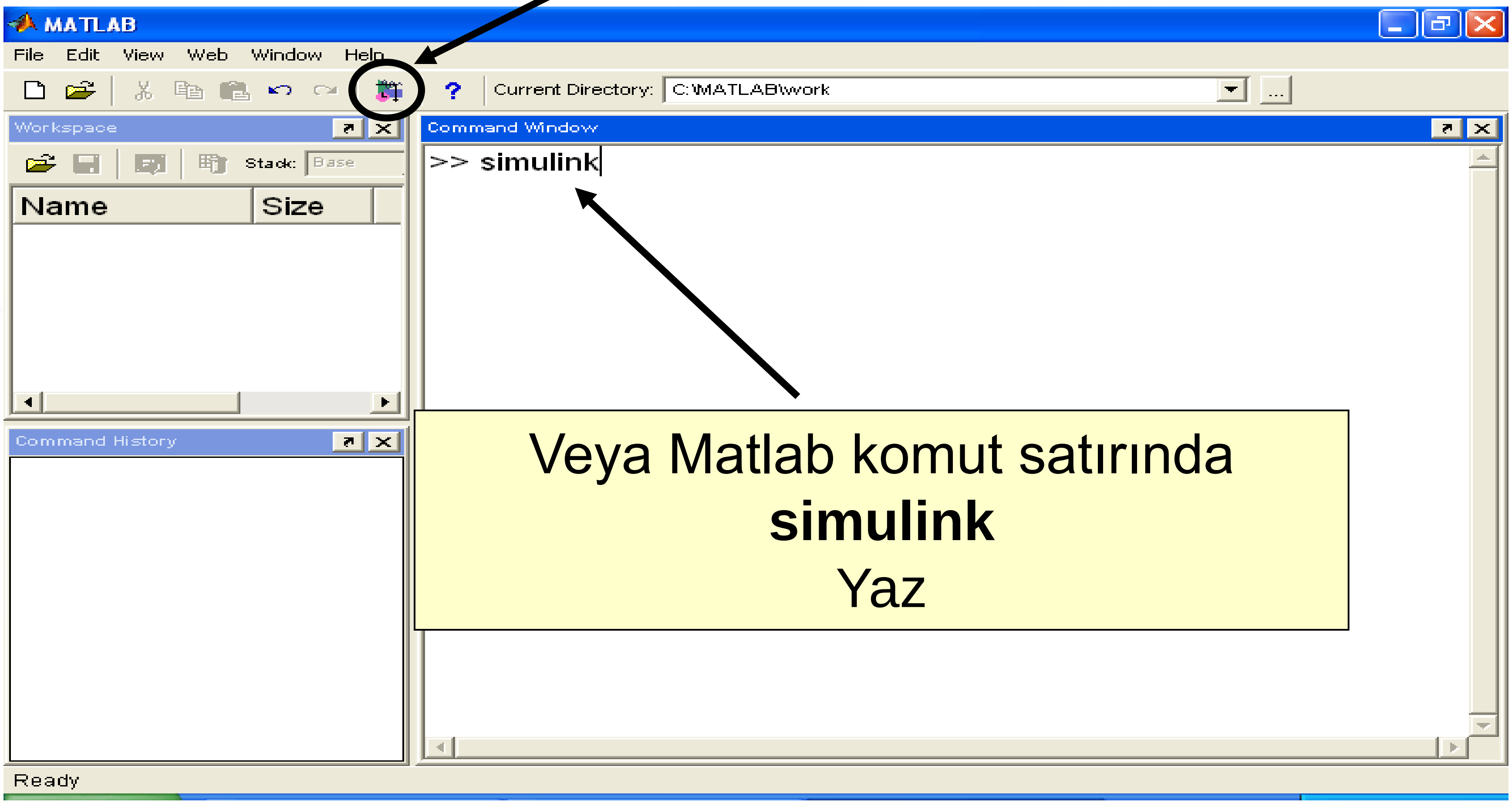
MATLAB

SIMULINK

Simulink Oturumunu Başlatma

SIMULINK icon üzerine tıkla



Veya Matlab komut satırında

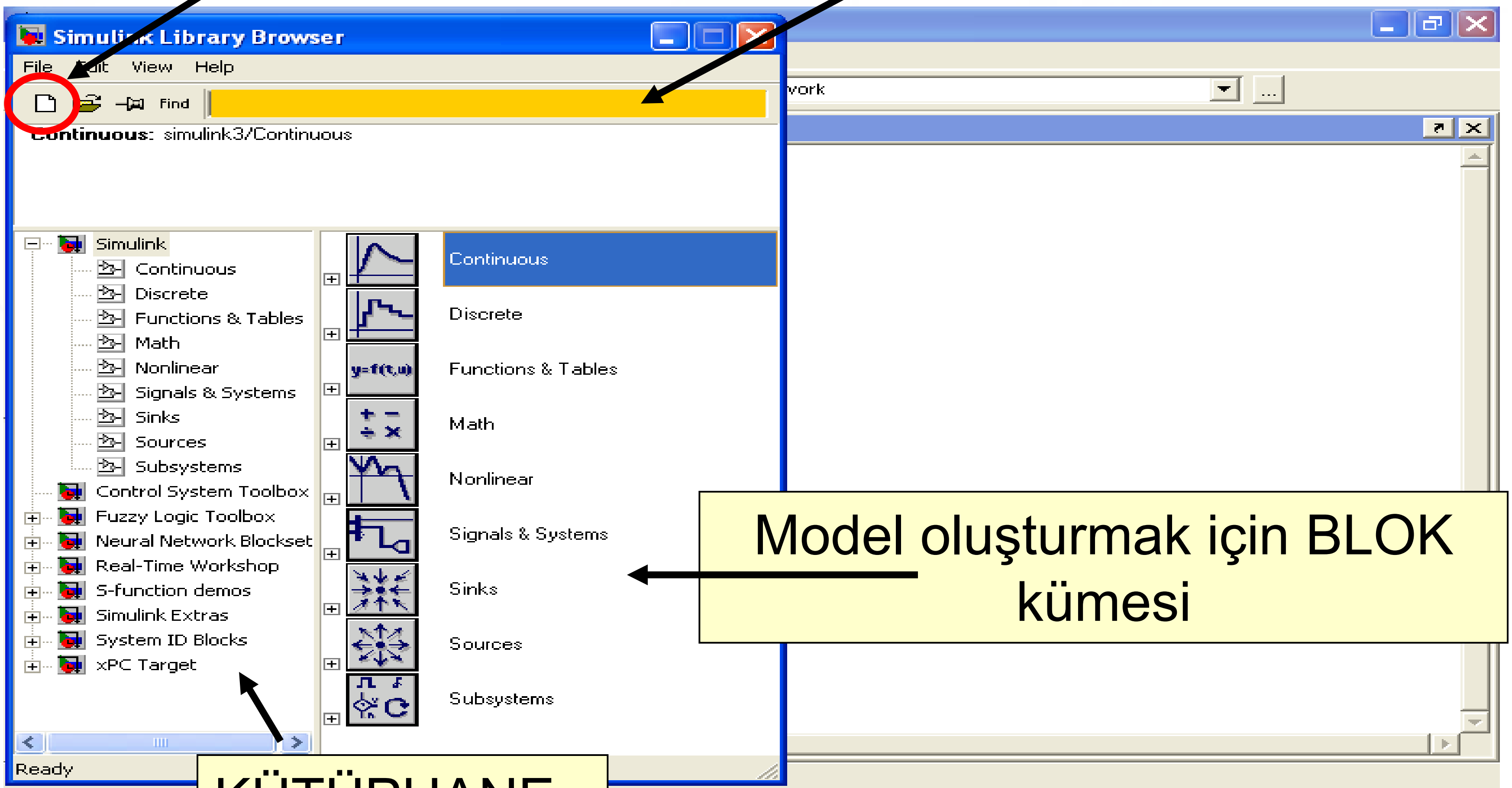
simulink

Yaz

Simulink Kütüphanesi

Yeni model iconu oluşturma

Arama penceresi

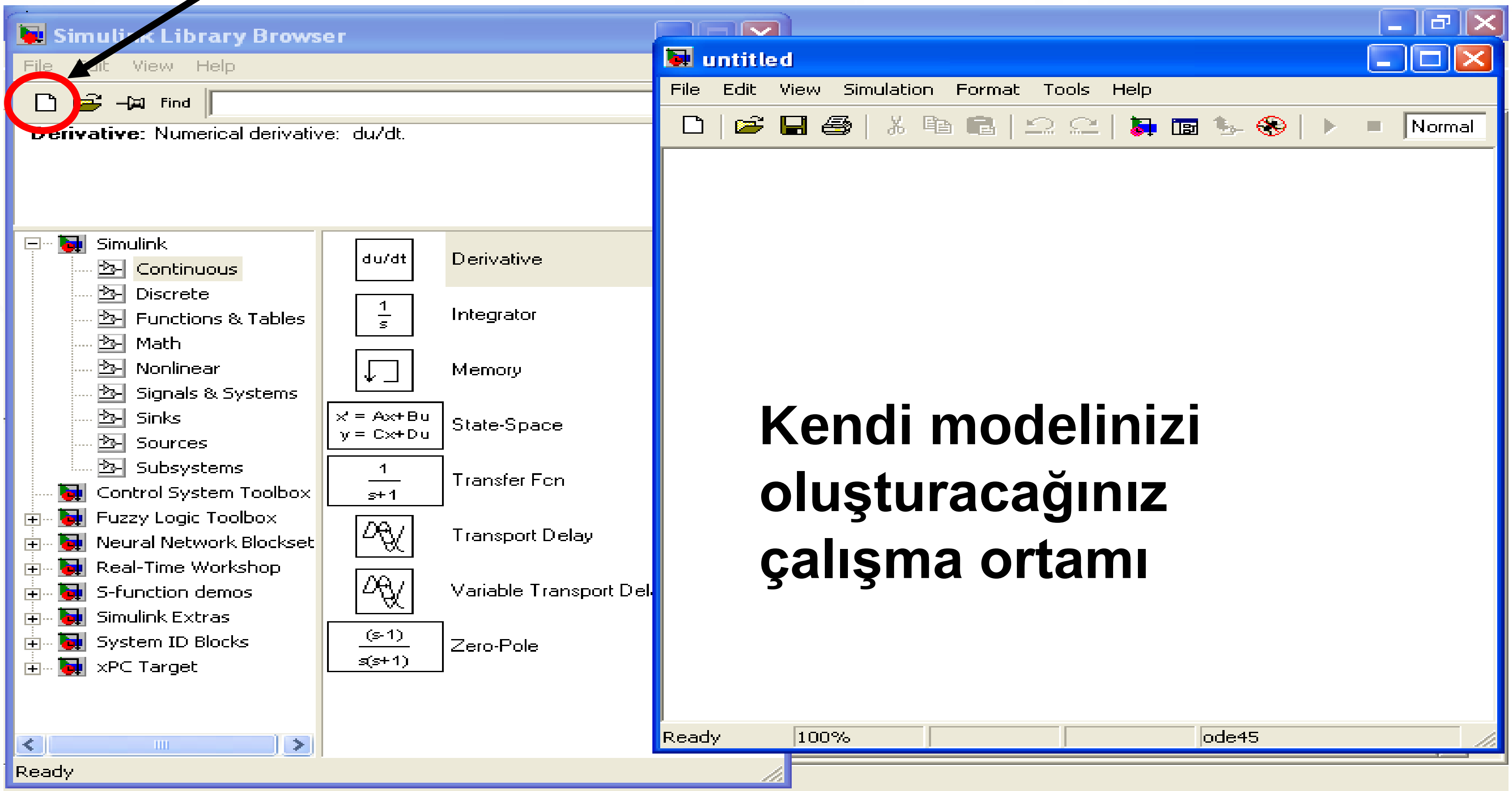


Model oluşturmak için BLOK kümesi

KÜTÜPHANE

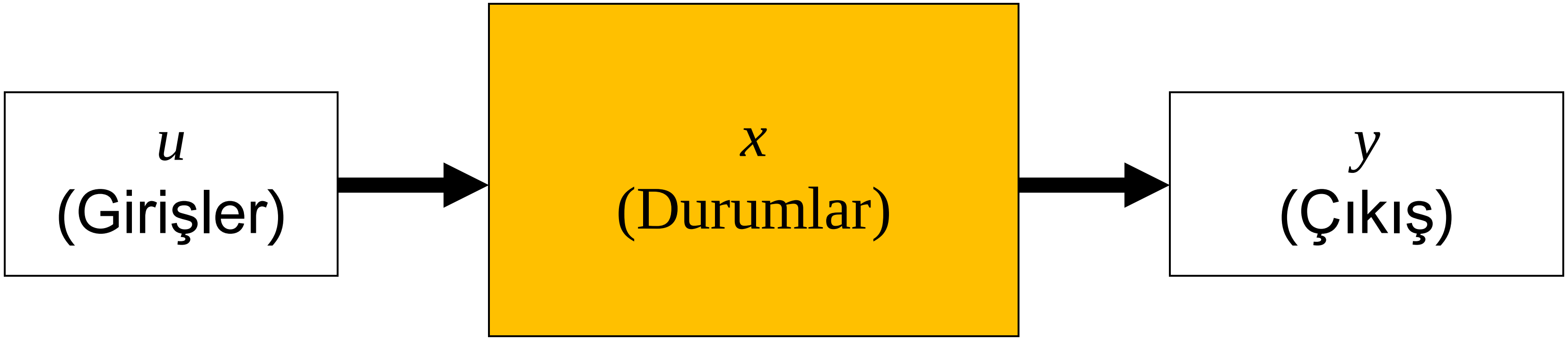
Yeni model oluşturma

Yeni model iconu oluşturma



Model Oluşturma

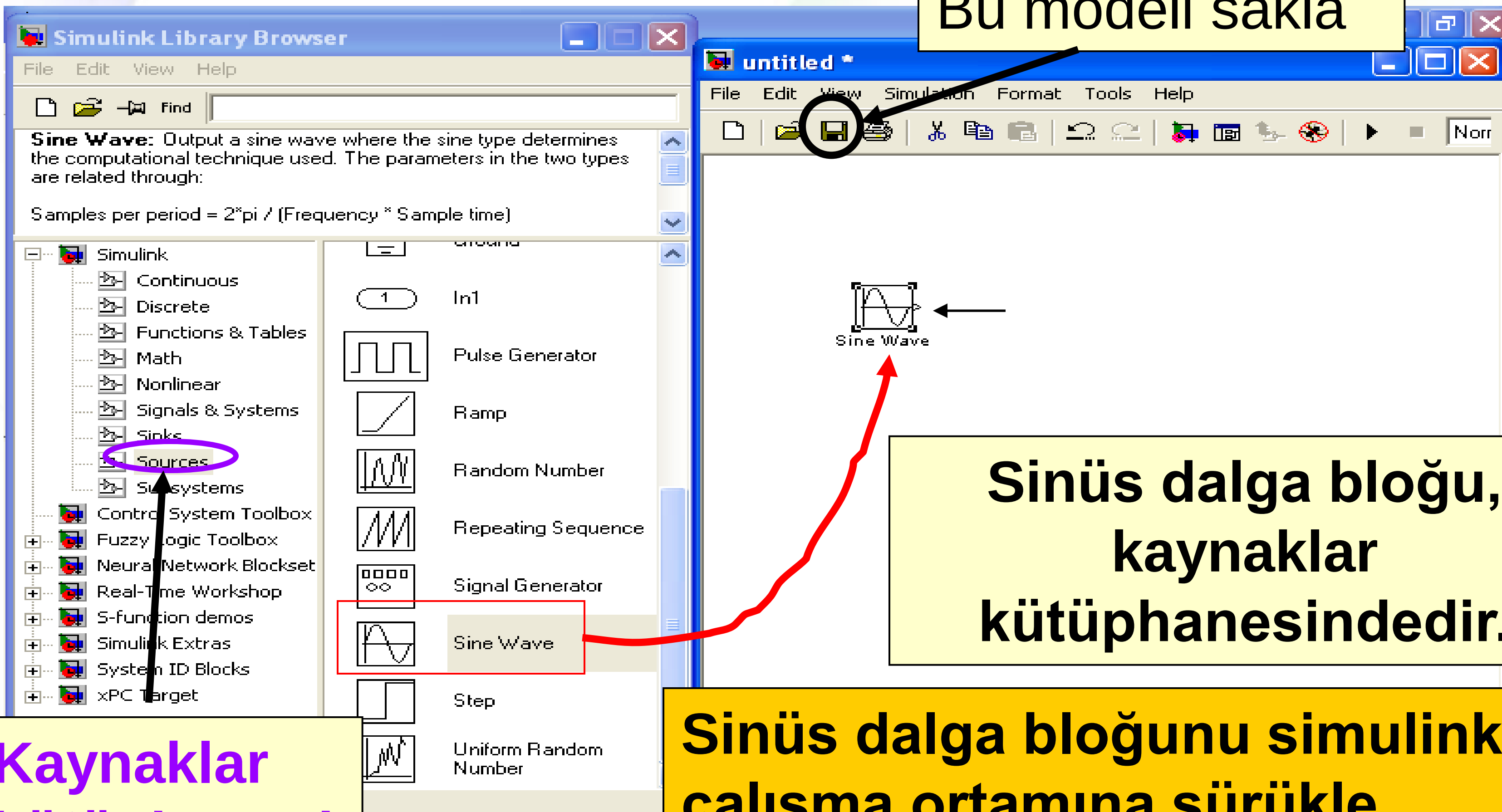
- **Simulink blok diyagramı – dinamik sistemin resimsel modeli**
- **Her blok ya sürekli yada ayrık çıkış üreten temel bir dinamik sistem gösterir.**
- **Doğrular blok çıkışlarına blok girişlerini bağlayan bağlantıları gösterir.**



Model Oluşturma (2)

- **Aşağıdaki adımlar bir sistem/model kurmak için size kılavuzluk edecektir:**
- **ADIM 1: Blokları oluşturma**
- **ADIM 2: Bağlantıları yapma**
- **ADIM 3: Parametreleri kurma**
- **ADIM 4: Simülasyonu çalıştırma**

Adım 1: Blokları oluşturma



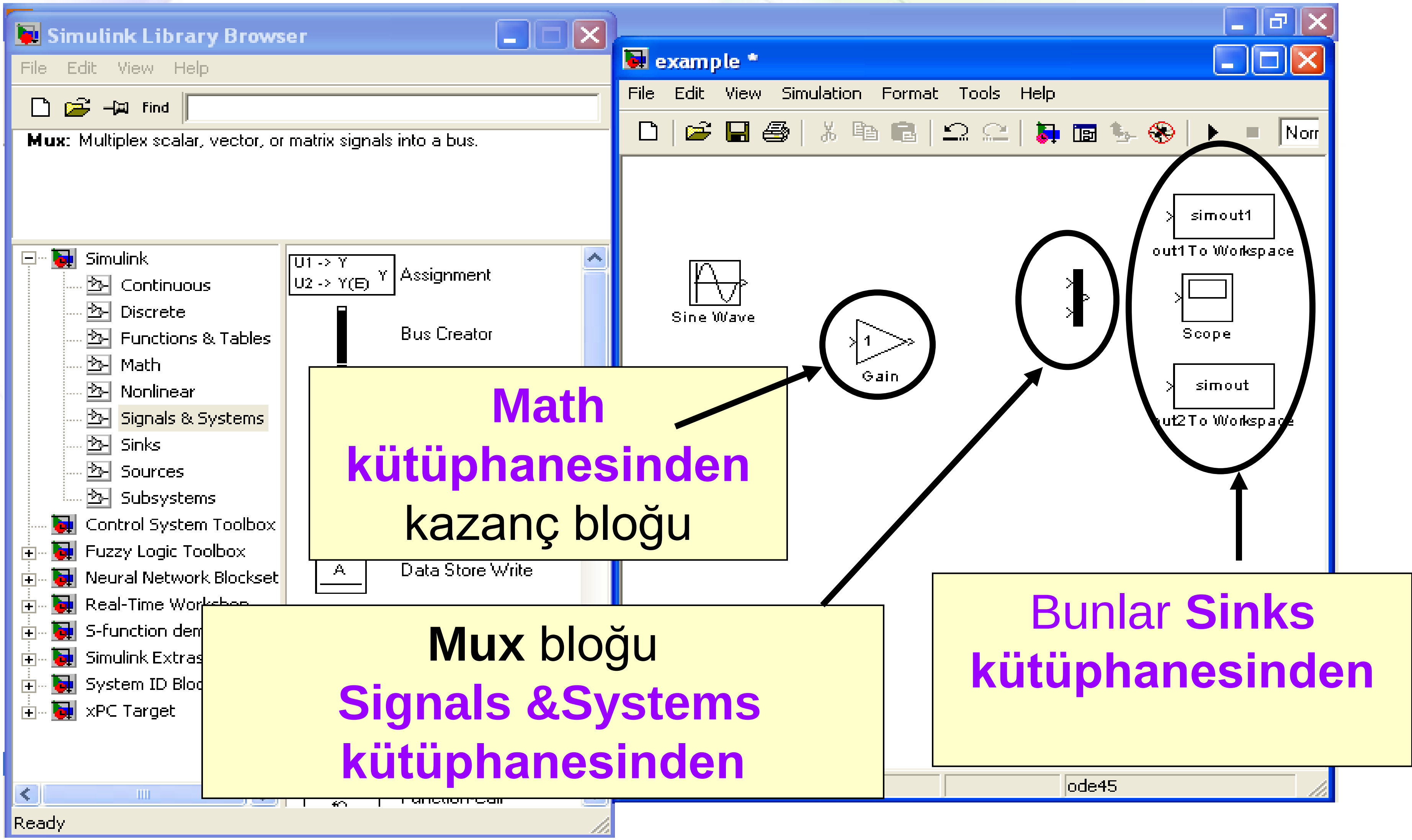
Bu modeli sakla

Sinüs dalga bloğu, kaynaklar kütüphanesindedir..

Kaynaklar kütüphanesi

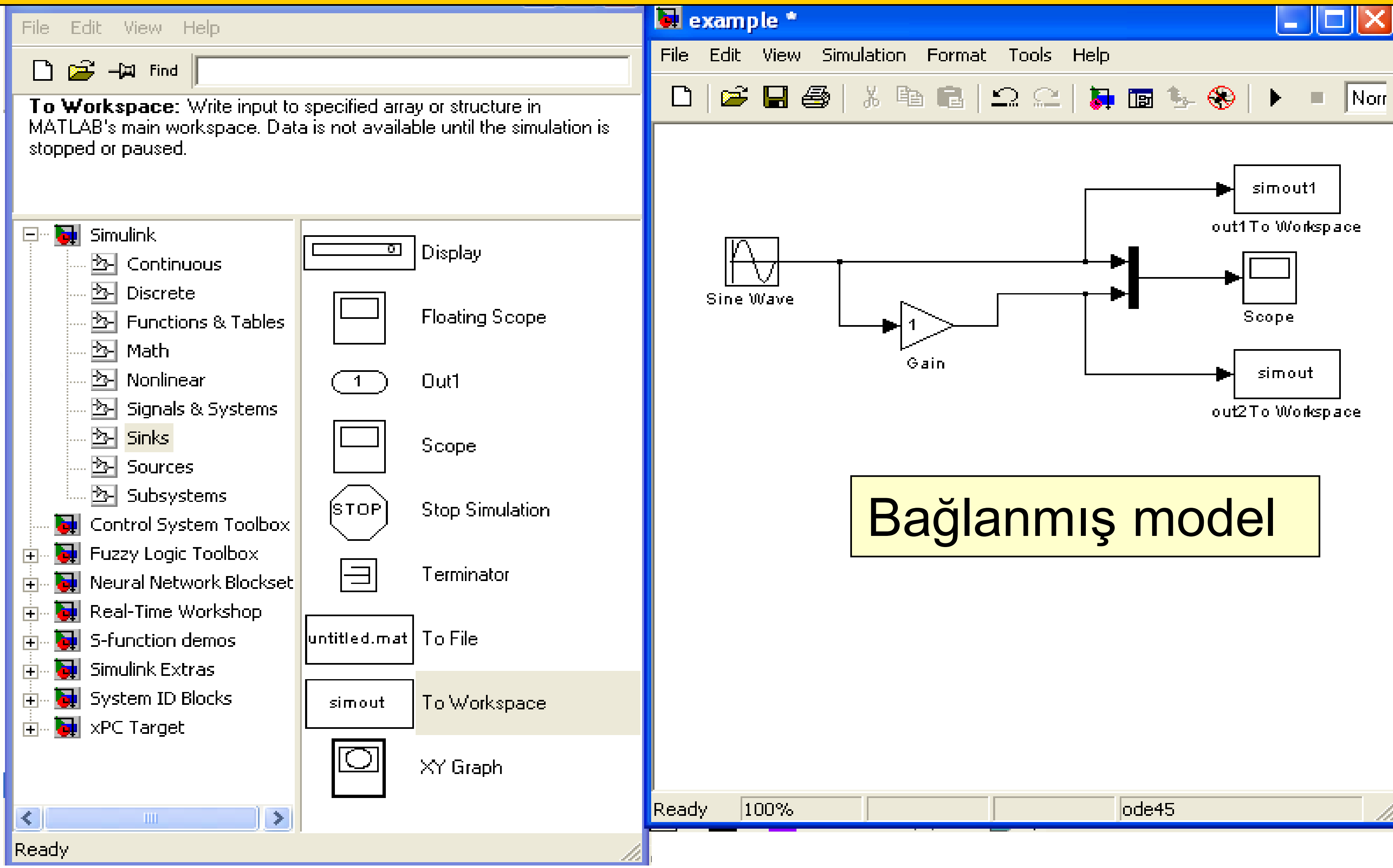
Sinüs dalga bloğunu simulink çalışma ortamına sürükle

Adım 1: Blokları oluşturma



Adım 2: Bağlantılar yapma

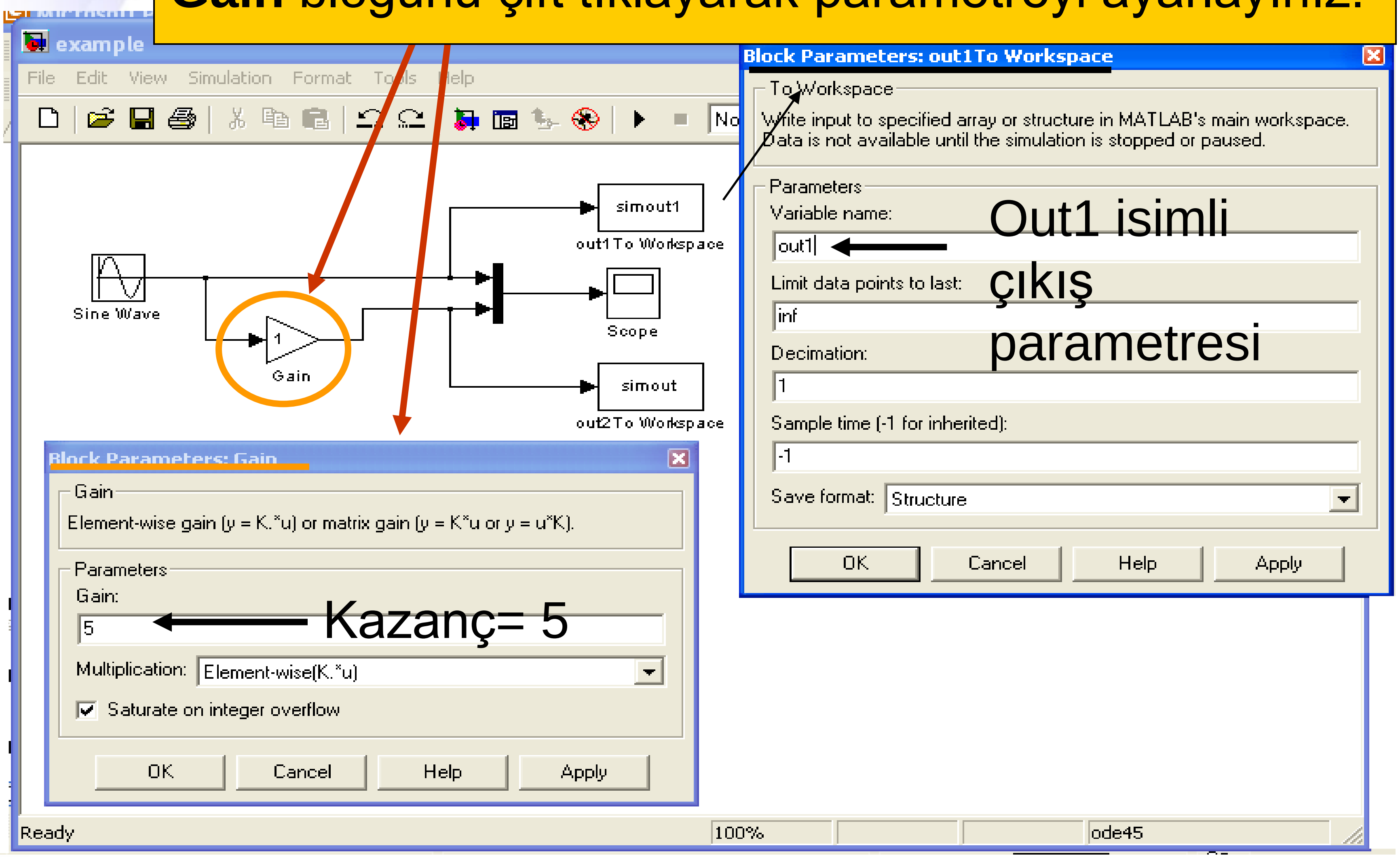
Bağlantı yapmak için: kaynak porttan gidilecek porta sürüklenir.



Bağlanmış model

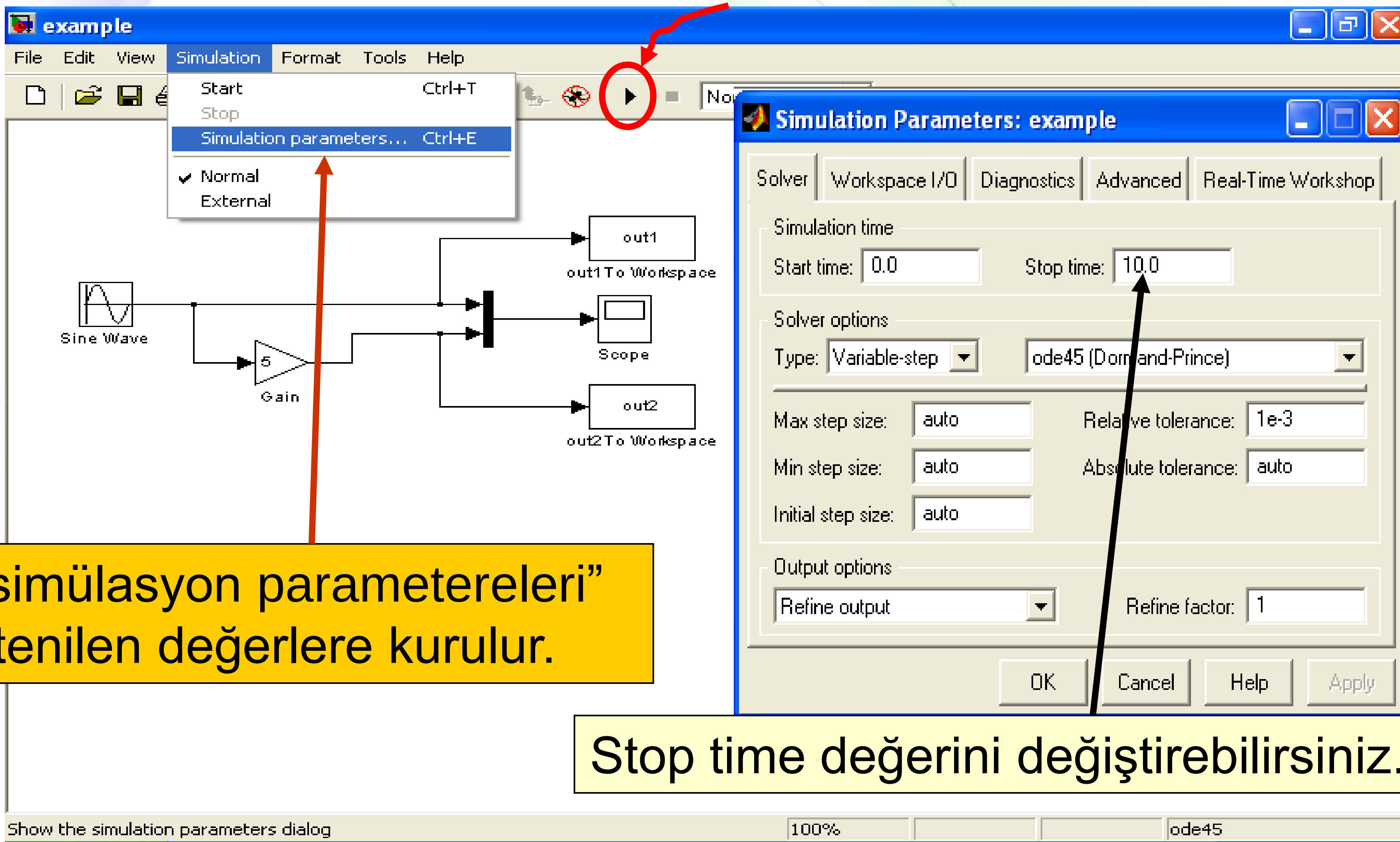
Adım 3: Parametre Kurma

Gain bloğunu çift tıklayarak parametreyi ayarlayınız.



Adım 4: Simülasyonu Çalıştırma

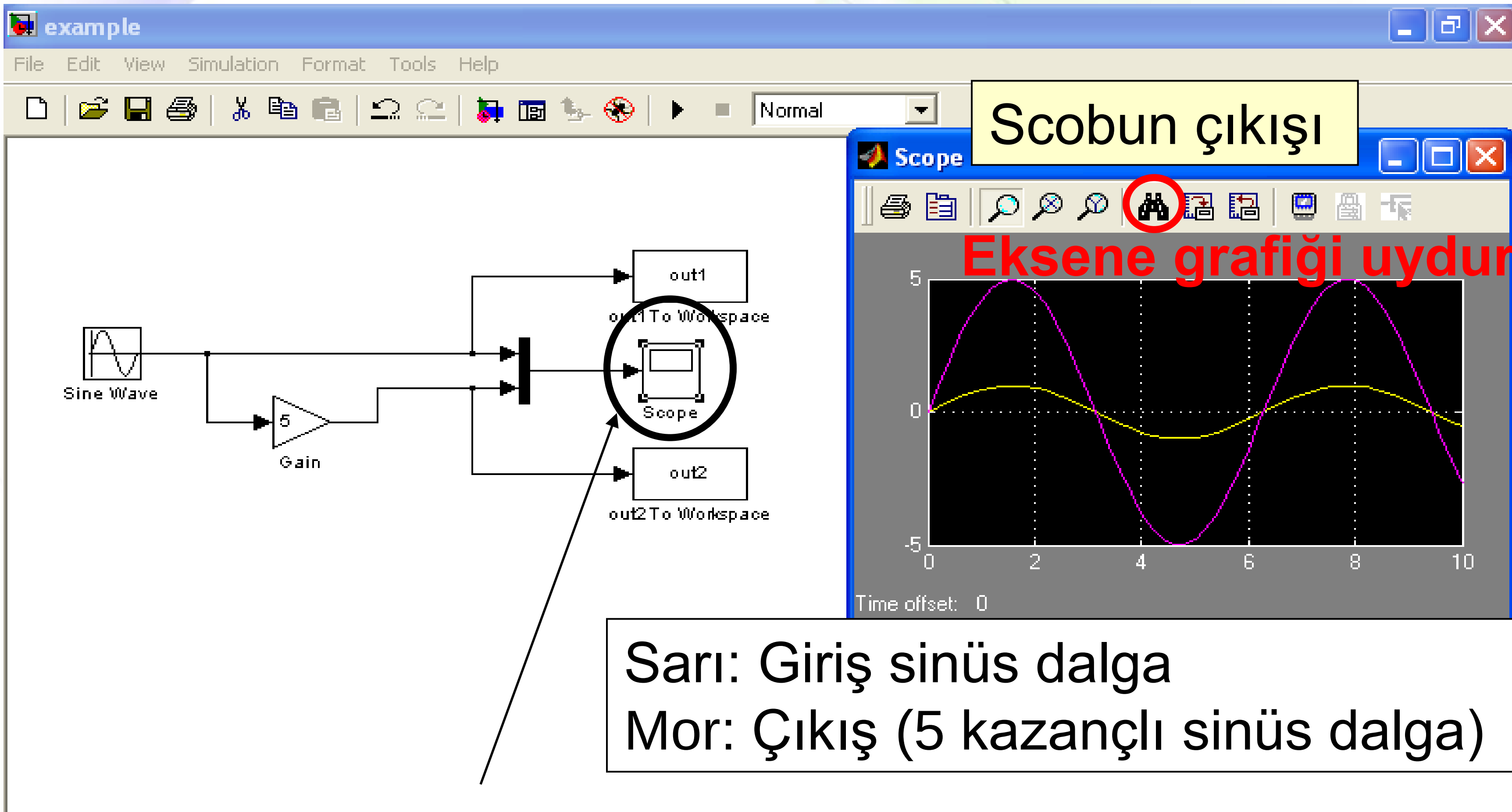
Simülasyonu çalıştırmak için run



"simülasyon parametereleri" istenilen değerlere kurulur.

Stop time değerini değiştirebilirsiniz.

Scope bloğu yoluyla çıkışa bakmak.



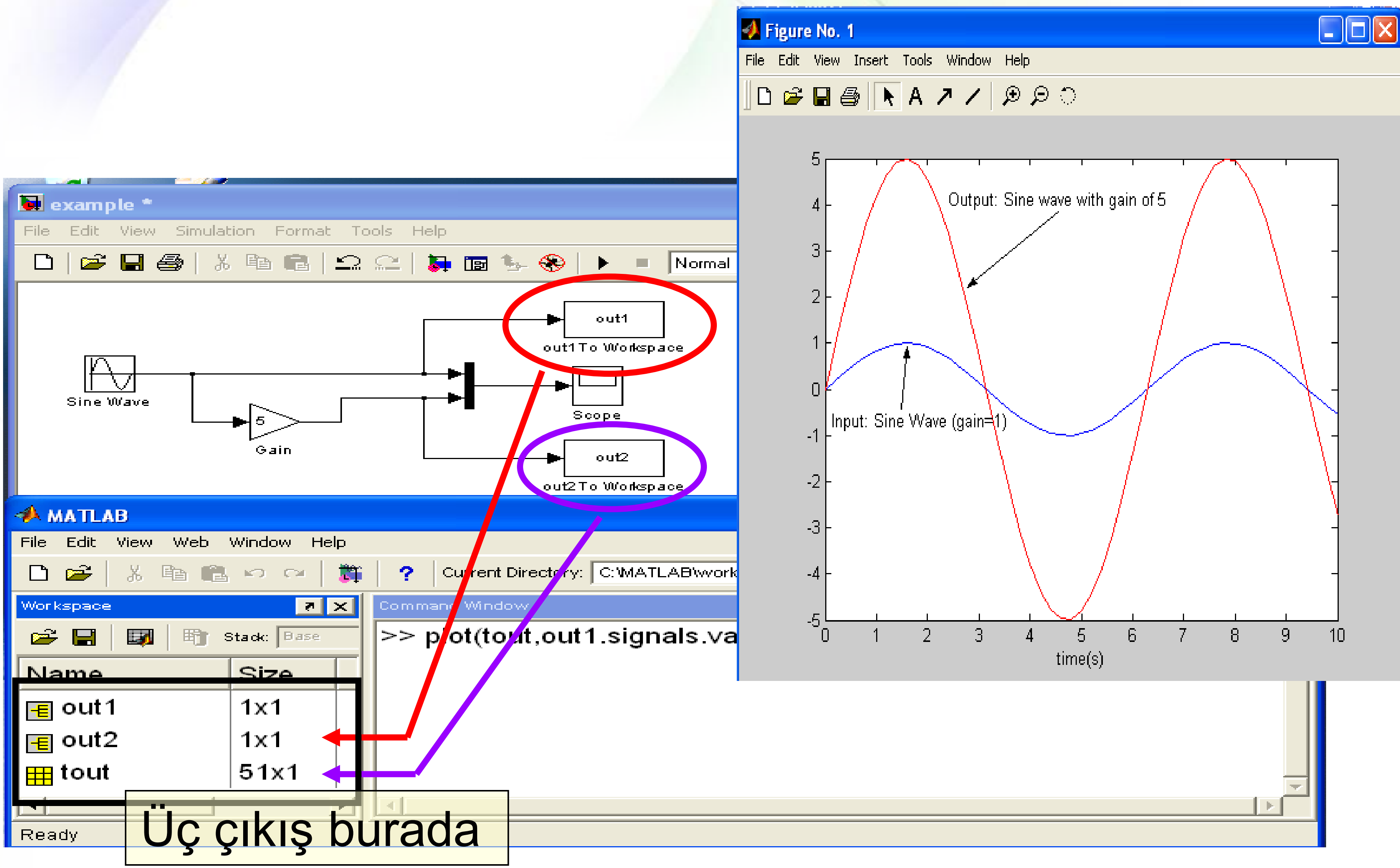
Scobun çıkışı

Eksene grafiği uydur.

Sarı: Giriş sinüs dalga
Mor: Çıkış (5 kazançlı sinüs dalga)

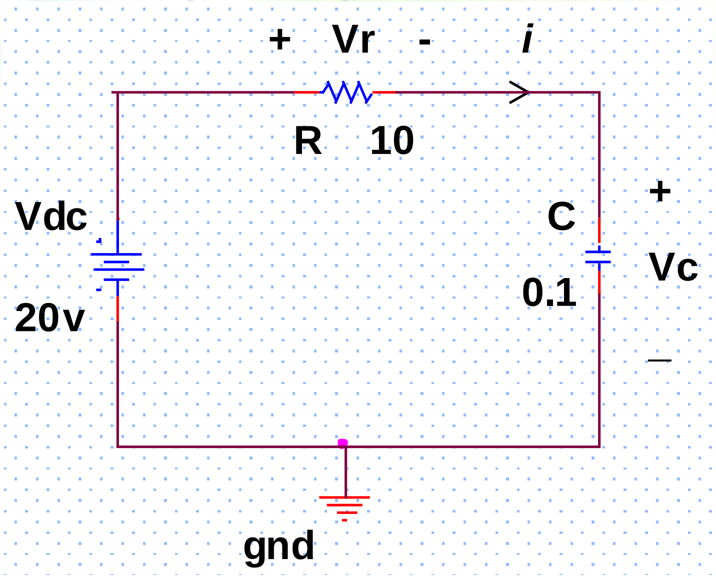
Double click on **Scope bloğu üzerine tıklayarak sonuca bak**.
Osiloskopa benziyor.

Çıkışa bak (*workspace*)



RC DEVRESİ

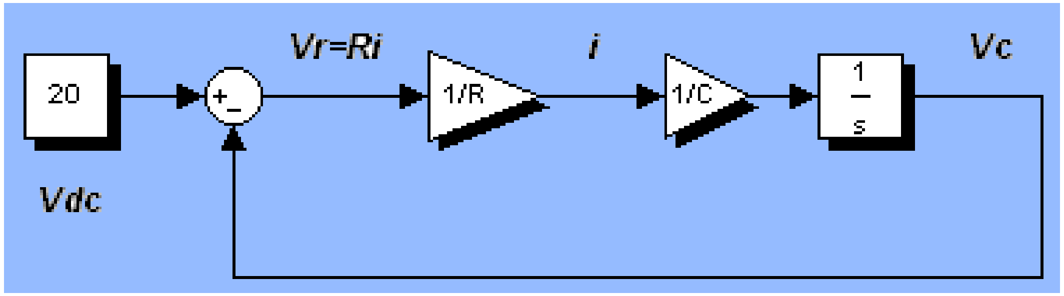
$$Vdc = Vr + Vc \quad \Rightarrow Vdc = Ri + Vc$$

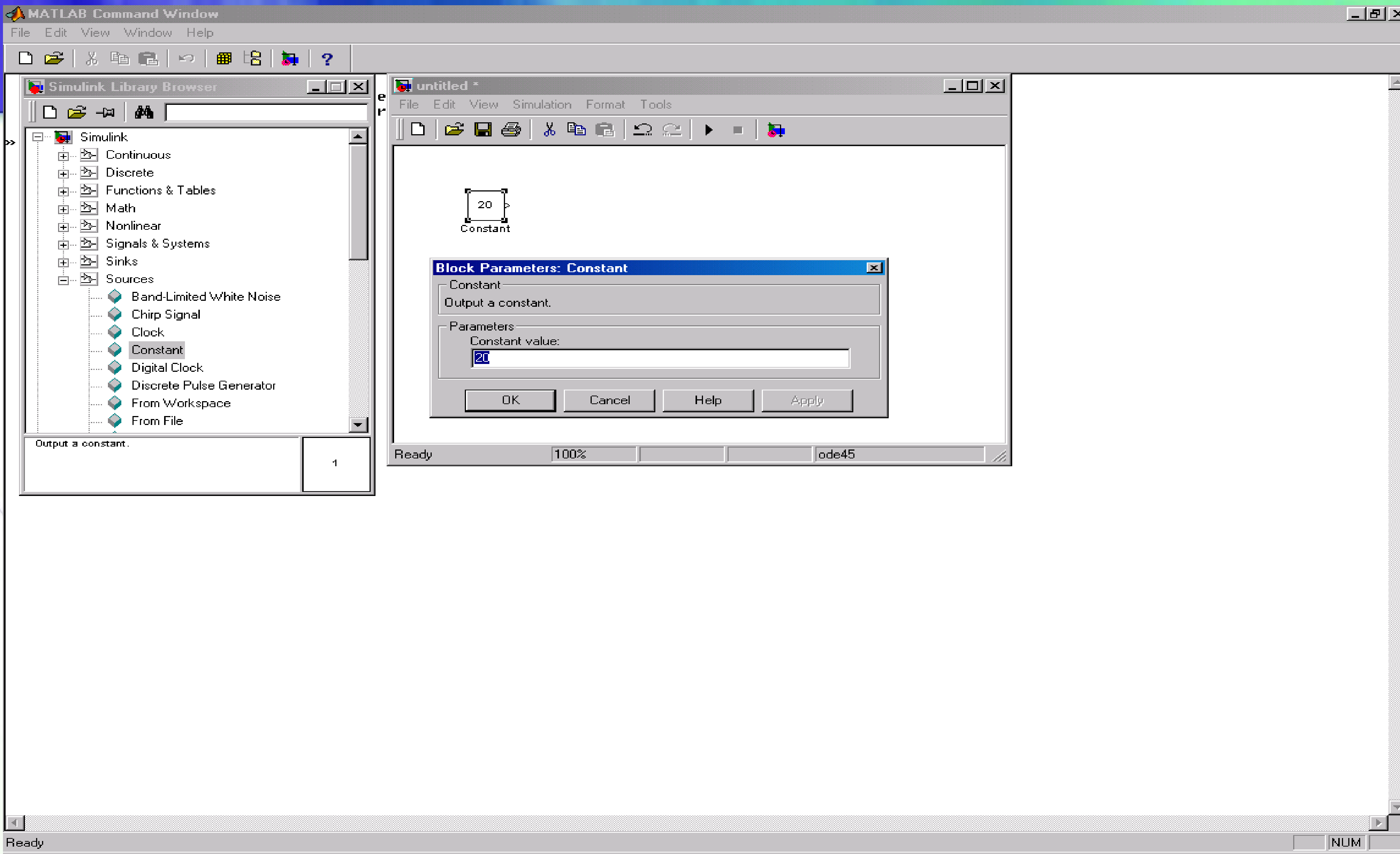


$$\Rightarrow Vdc - Vc = Ri = Vr$$

$$\Rightarrow i = \frac{Vdc - Vc}{R}$$

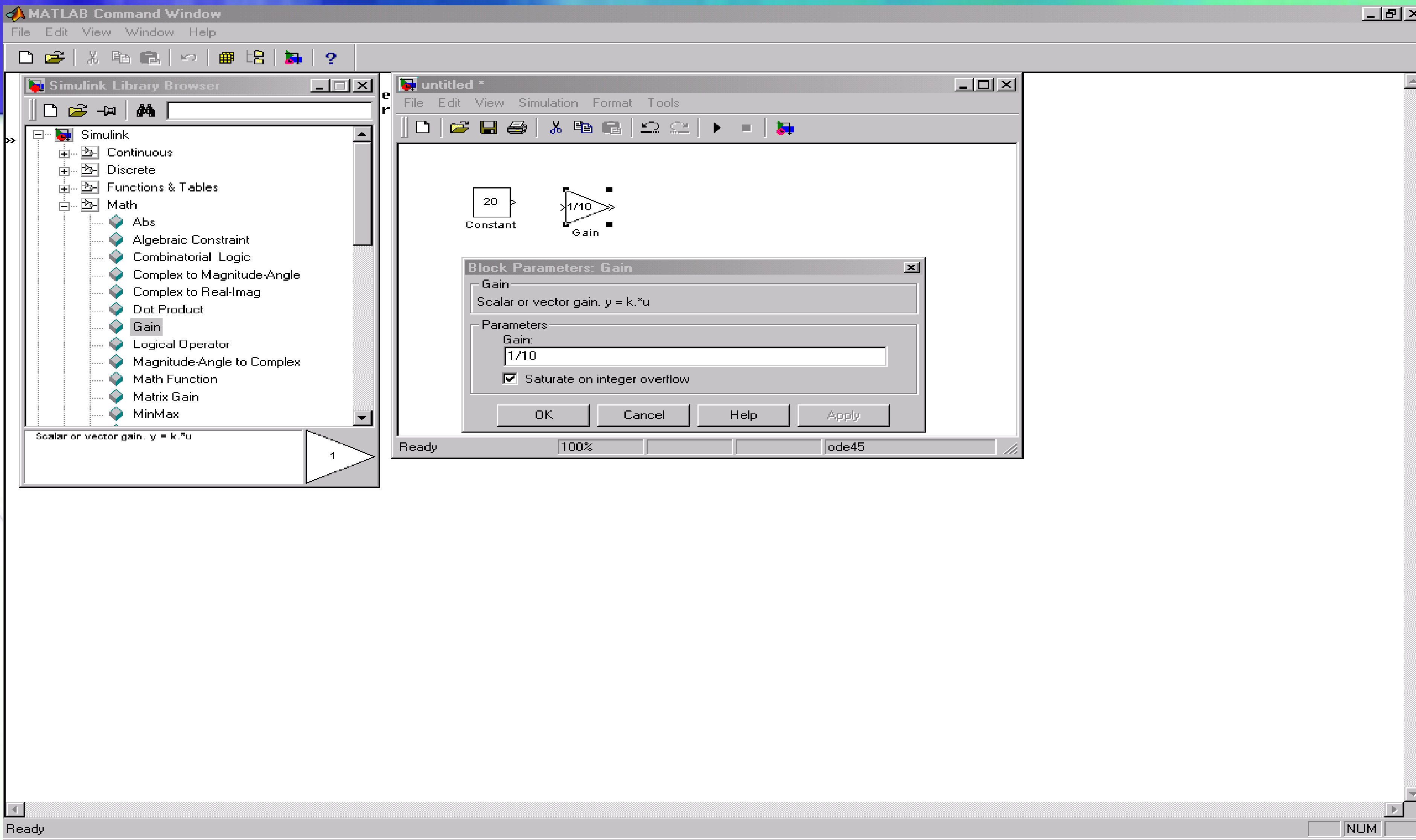
$$Vc = \frac{1}{C}\int i(dt)$$





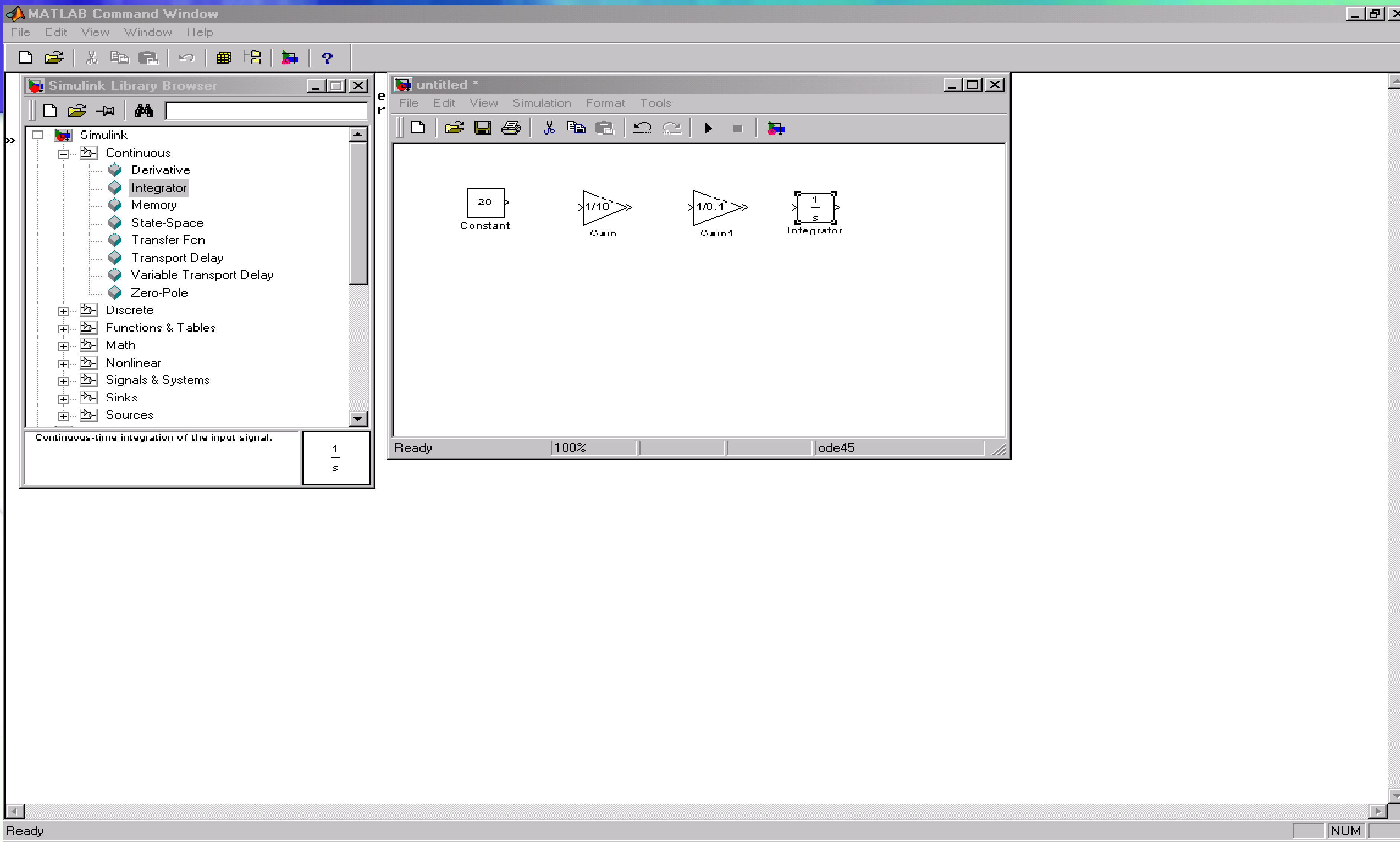
❖CONSTANT BLOĞA DEĞER ATANMASI

Matlab & Simulink

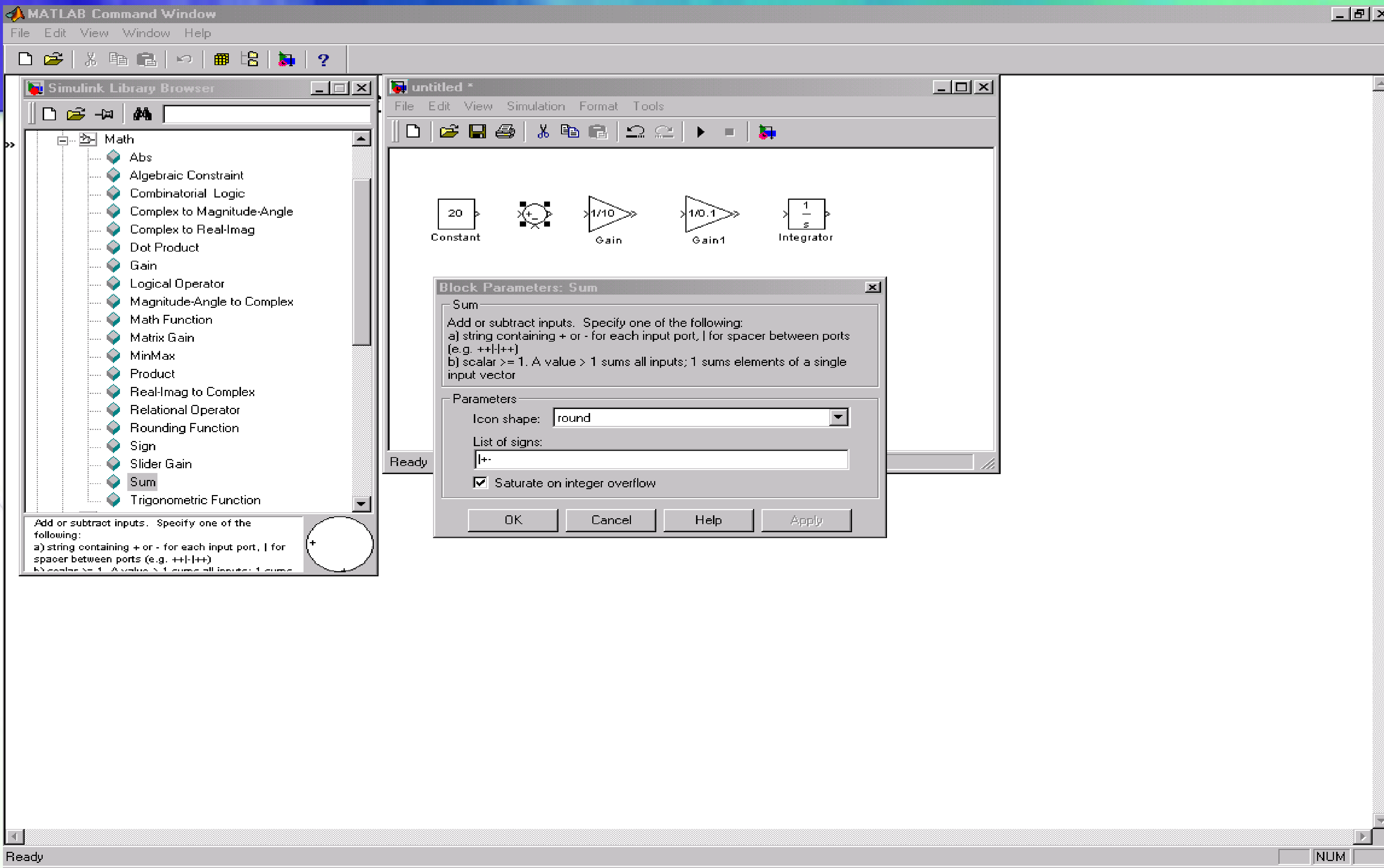


❖KAZANCIN (GAİN) AYARLANMASI

Matlab & Simulink

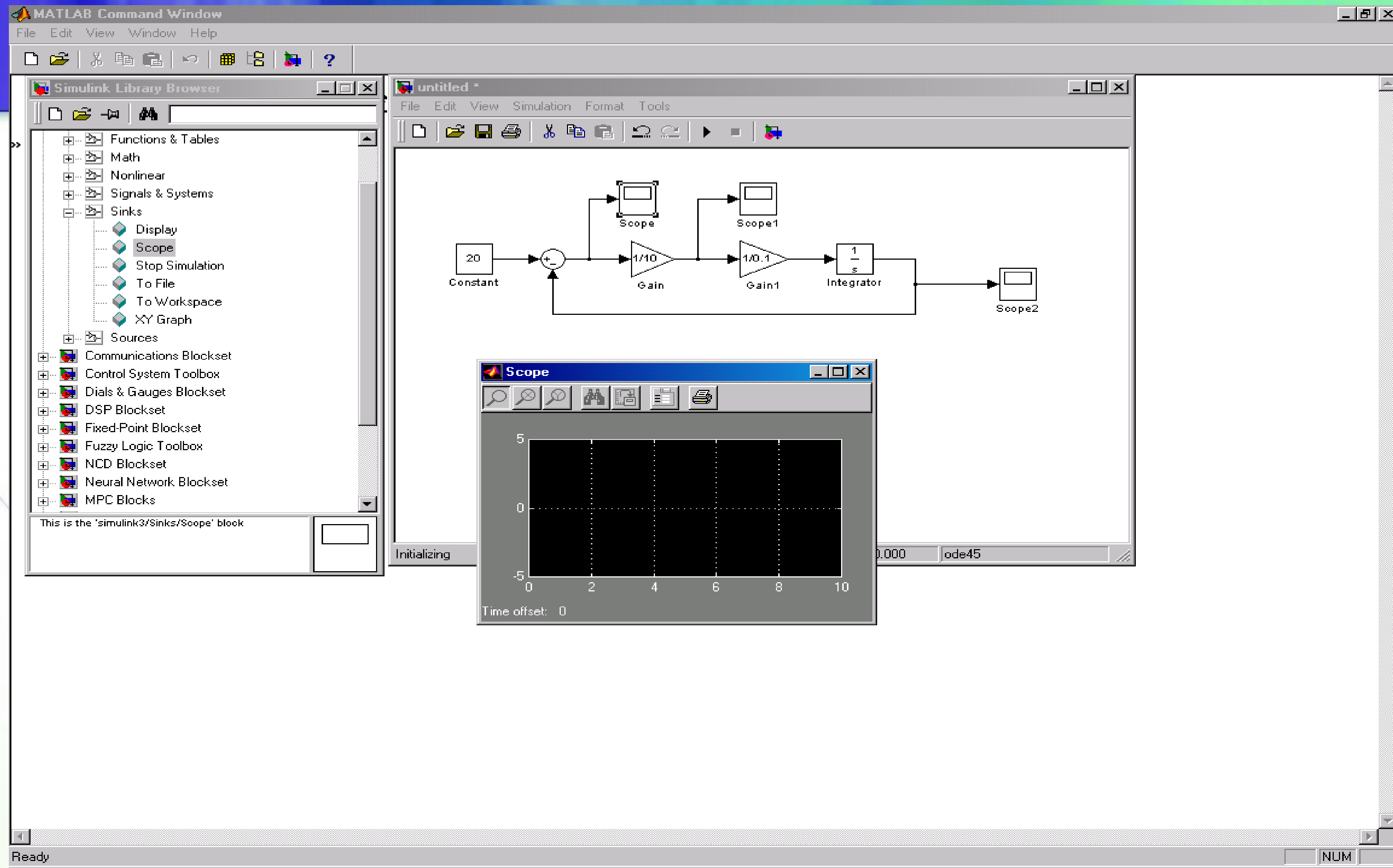


❖İNTEGRATÖR YERLEŞTİRİLMESİ

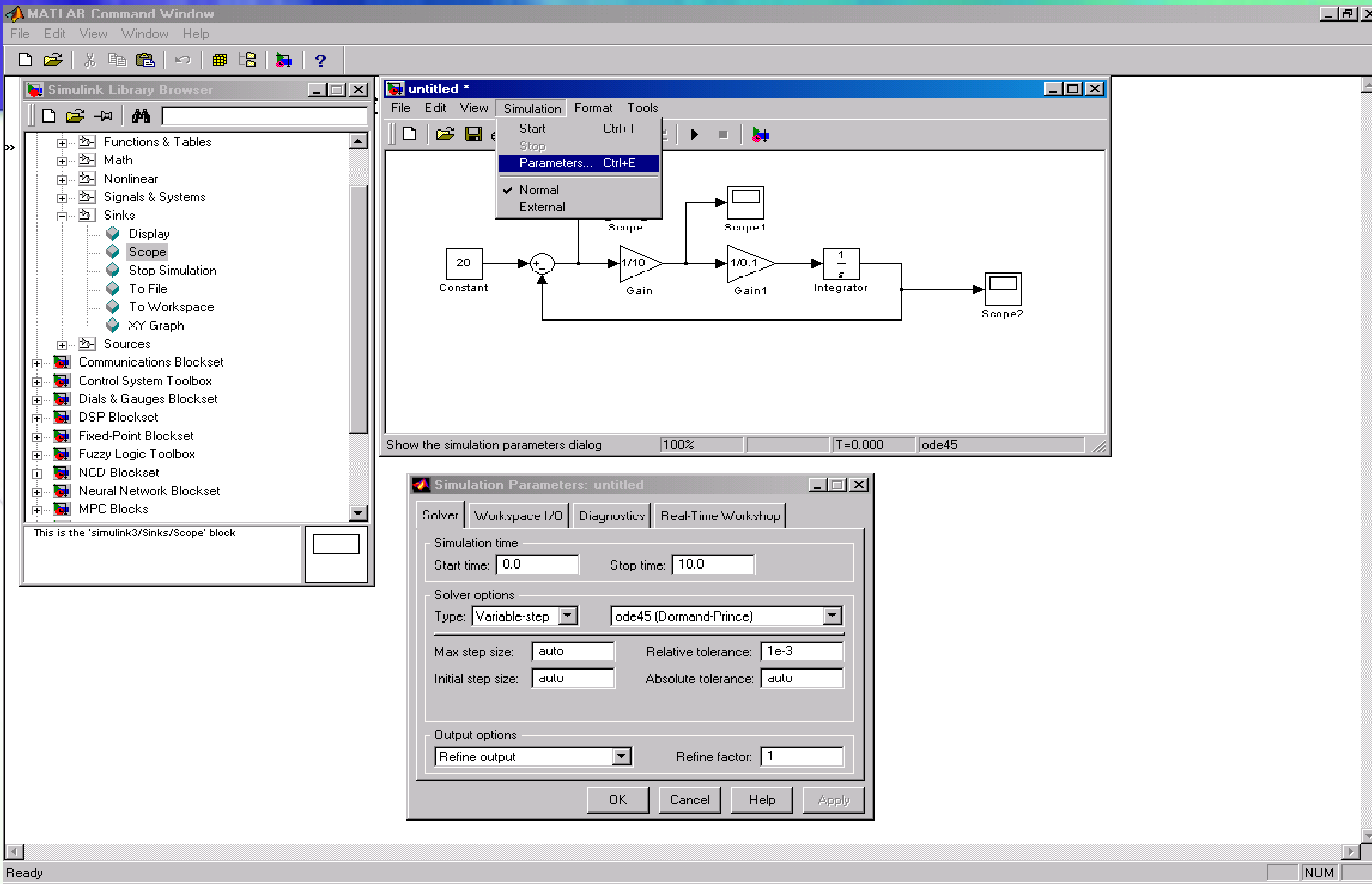


❖TOPLAYICININ (SUM) AYARLANMASI

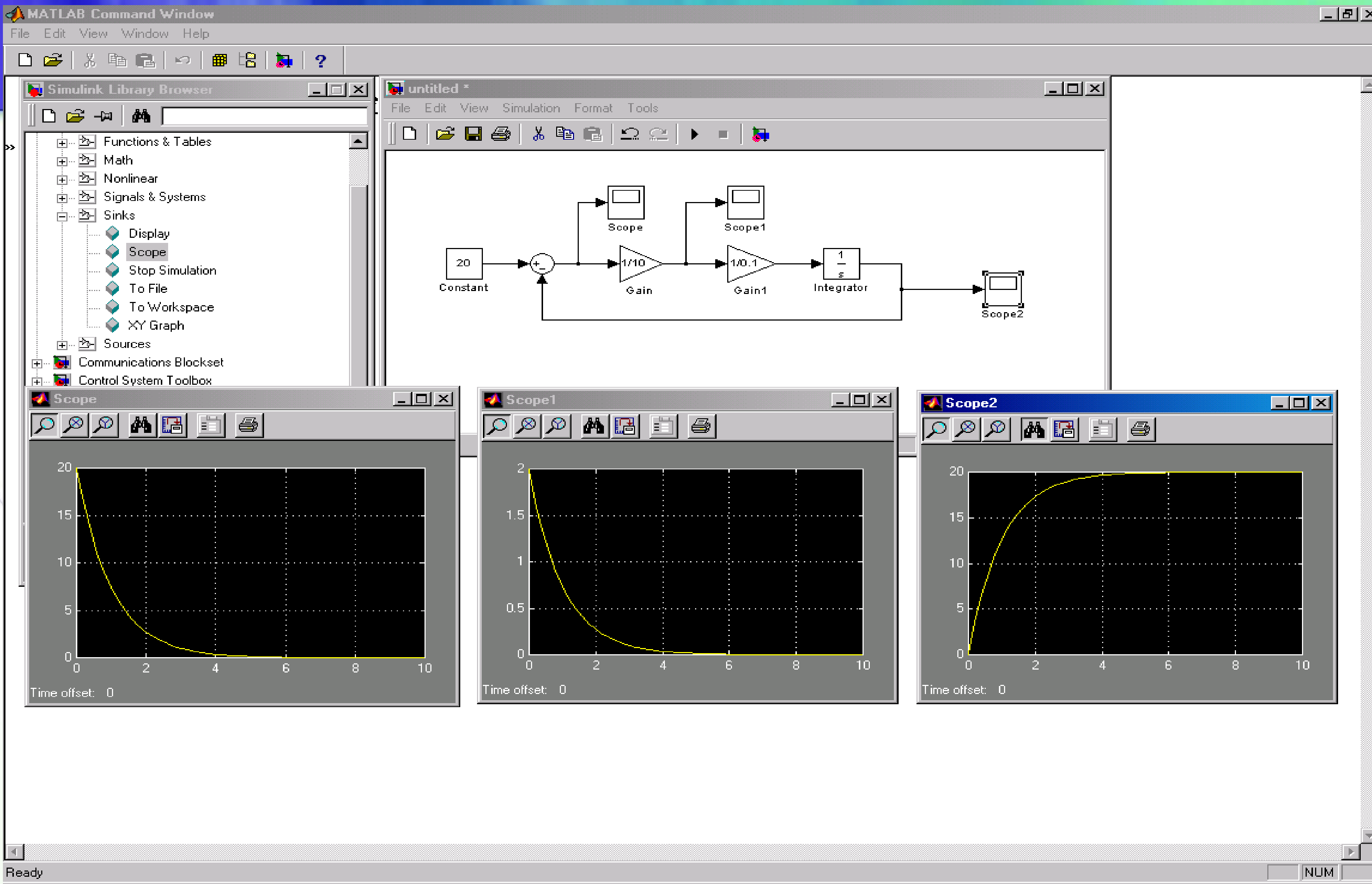
Matlab & Simulink



❖SCOPELARIN BAĞLANMASI VE SCOPE EKRANI



SİMULASYON PARAMETRELERİNİN AYARLANMASI



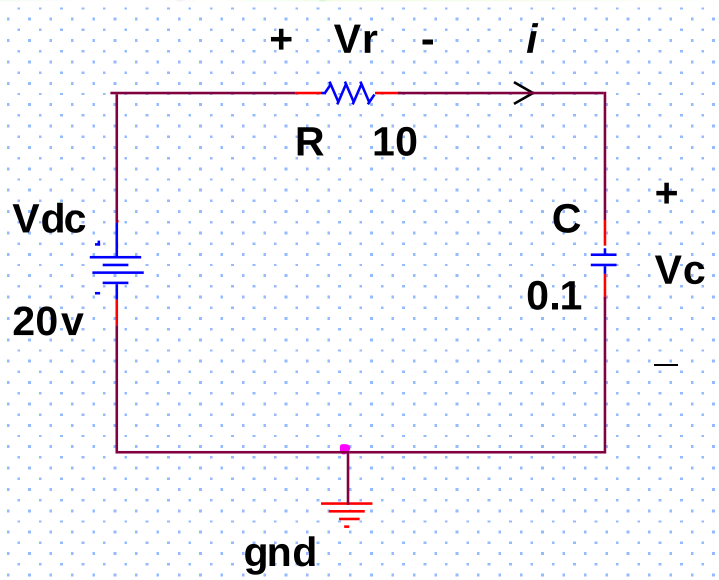
❖SİMÜLASYON SONUÇLARI (DİRENÇ GERİLİMİ, AKIM VE KAPASİTE GERİLİMİ)

Matlab & Simulink

➤RC DEVRESİ

$$Vdc = Vr + Vc \ldots (1)$$

$$i = C\frac{dVc}{dt} \ldots (2)$$



$$Vdc = Ri + Vc$$

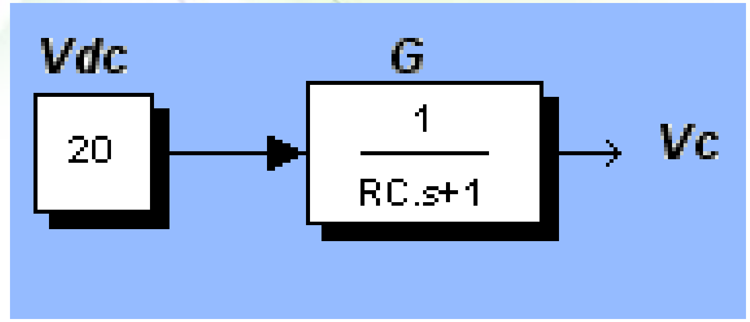
$$\Rightarrow Vdc = RC\frac{dVc}{dt} + Vc$$

$$Vdc(s) = sRCVc(s) + Vc(s) \Rightarrow Vc(s) = \frac{Vdc(s)}{1 + sRC}$$

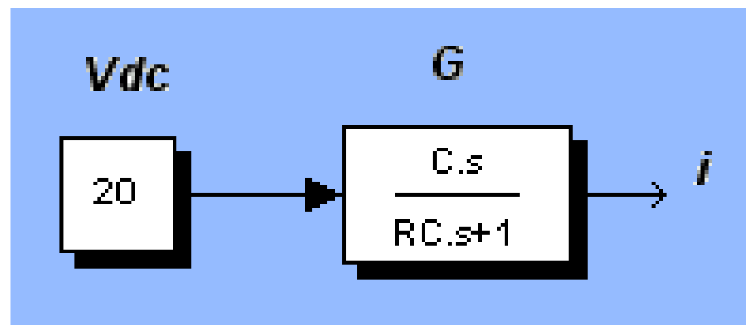
$$i(s) = \frac{sC}{1 + sRC}Vdc(s)$$

Matlab & Simulink

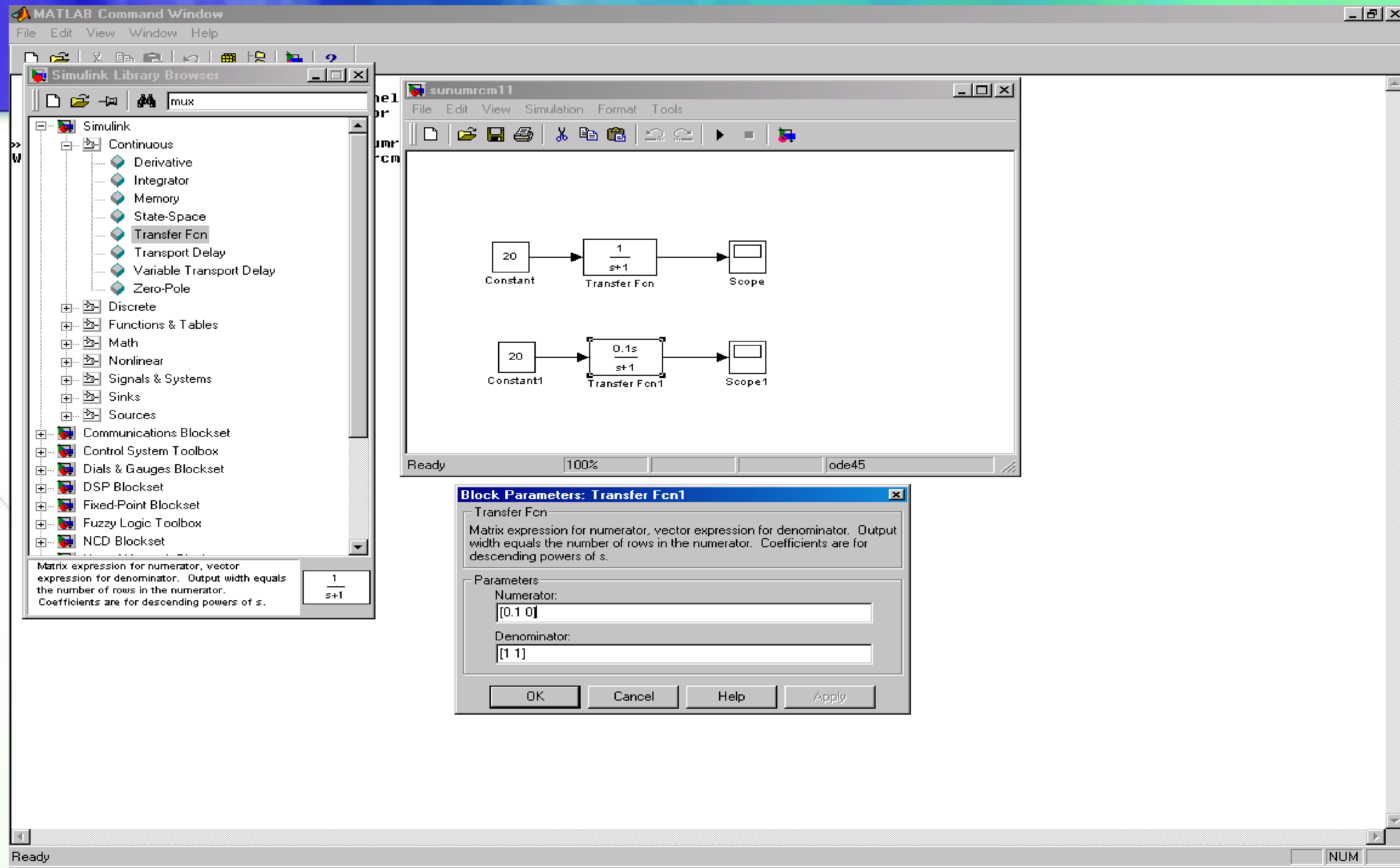
$$Vc(s) = \frac{1}{1 + sRC} Vdc(s)$$



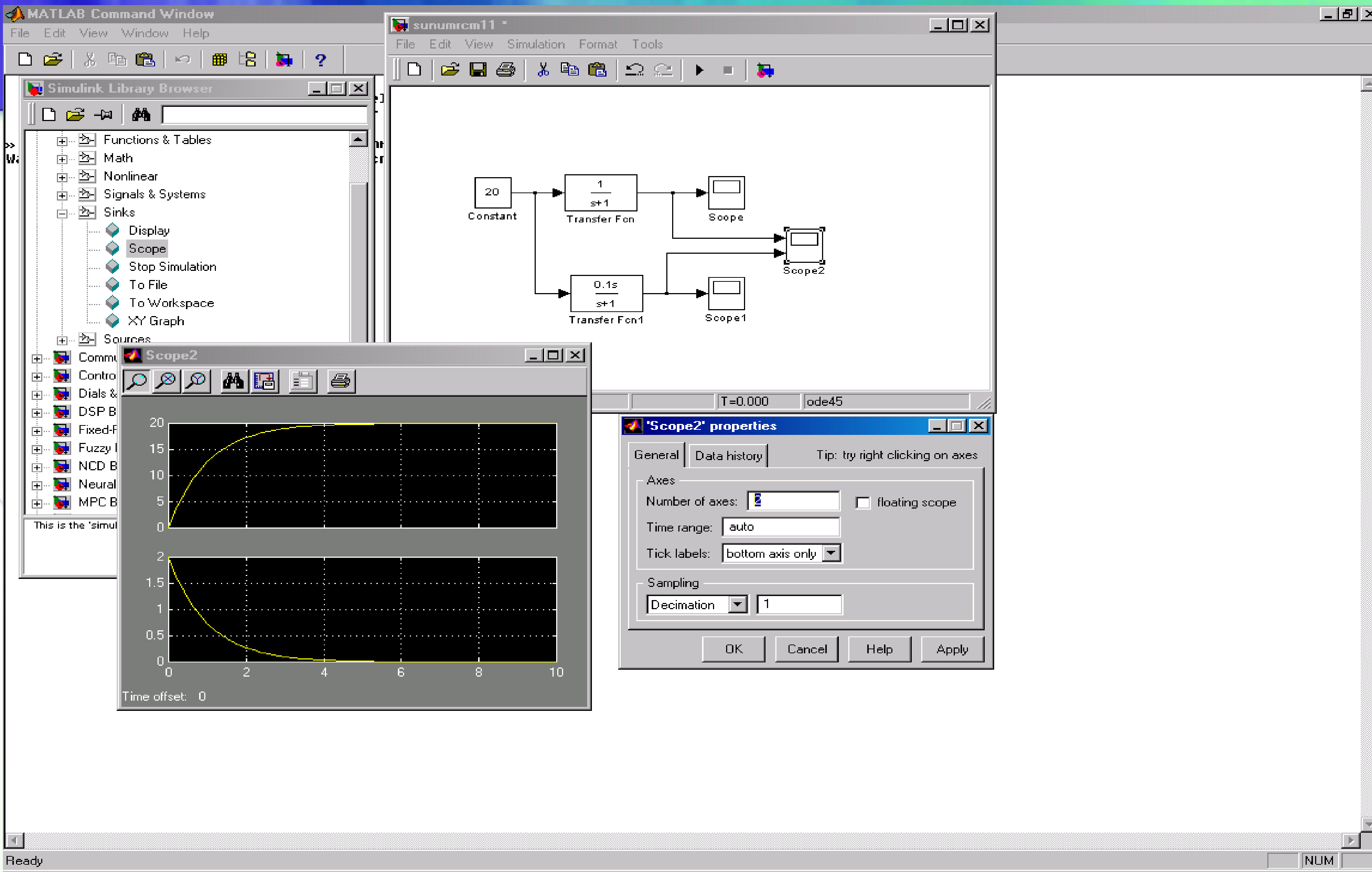
$$i(s) = \frac{sC}{1 + sRC} Vdc(s)$$



Matlab & Simulink

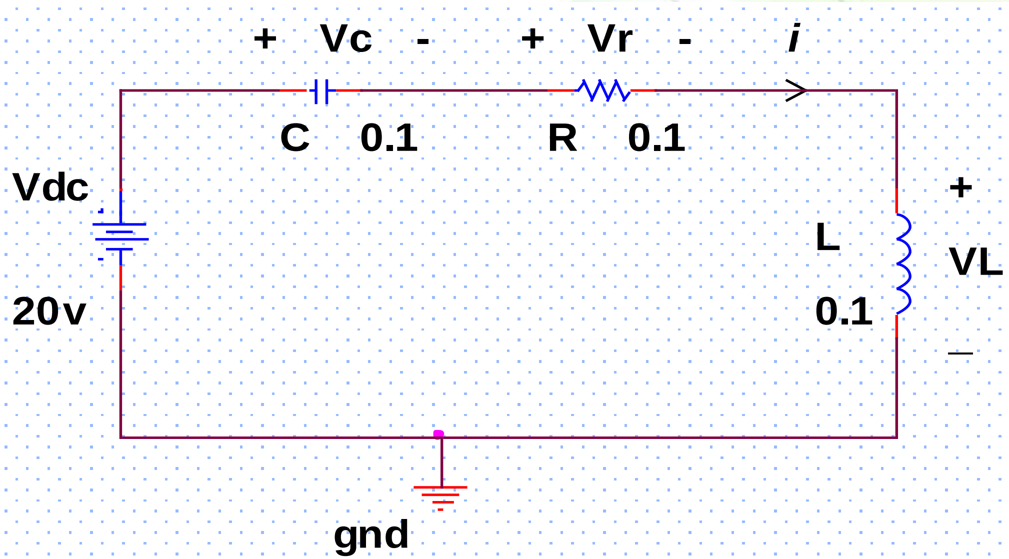


❖ TRANSFER FUNCTION PARAMETRELERİNİN GİRİLMESİ



❖SCOPE EKSEN SAYISININ ARTTIRILMASI

➢RLC DEVRESİ



$$Vdc = Vr + Vc + VL \qquad i = C\frac{dVc}{dt} \qquad V_L = L\frac{di}{dt}$$

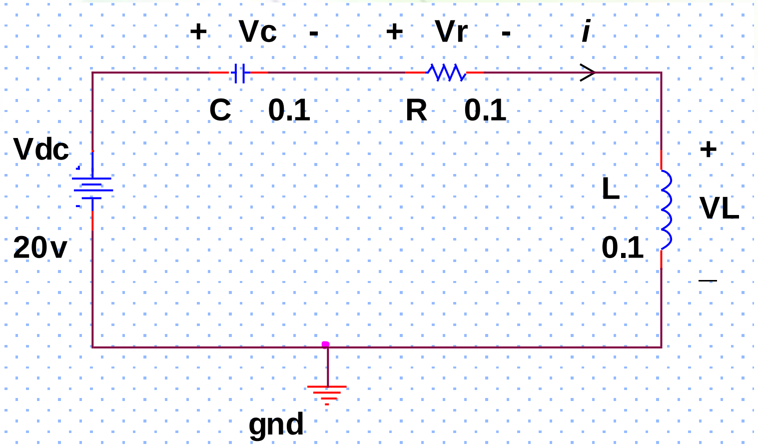
$$Vdc = Ri + L\frac{di}{dt} + Vc$$

Matlab & Simulink

➢RLC DEVRESİ

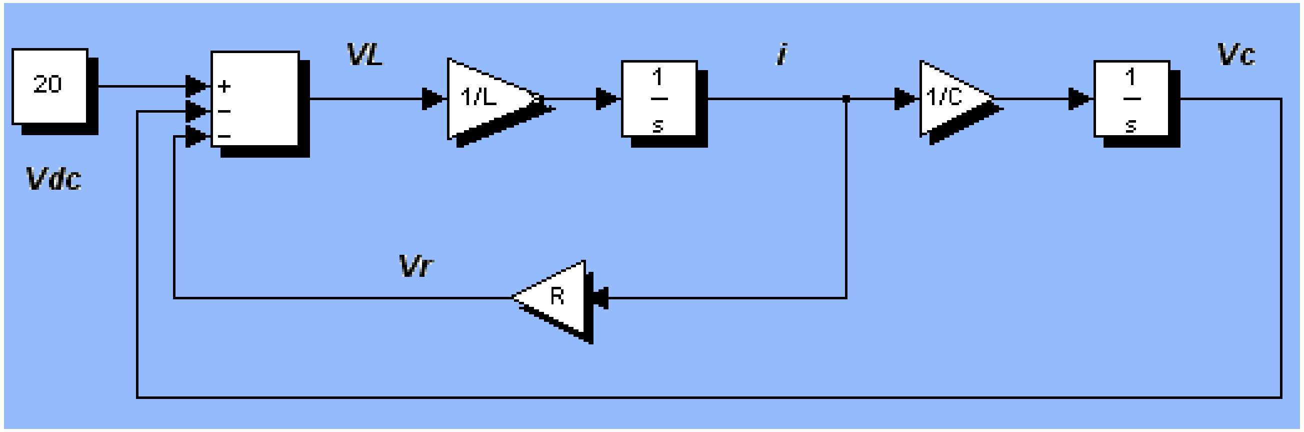
$$Vdc - Vr - Vc = VL$$

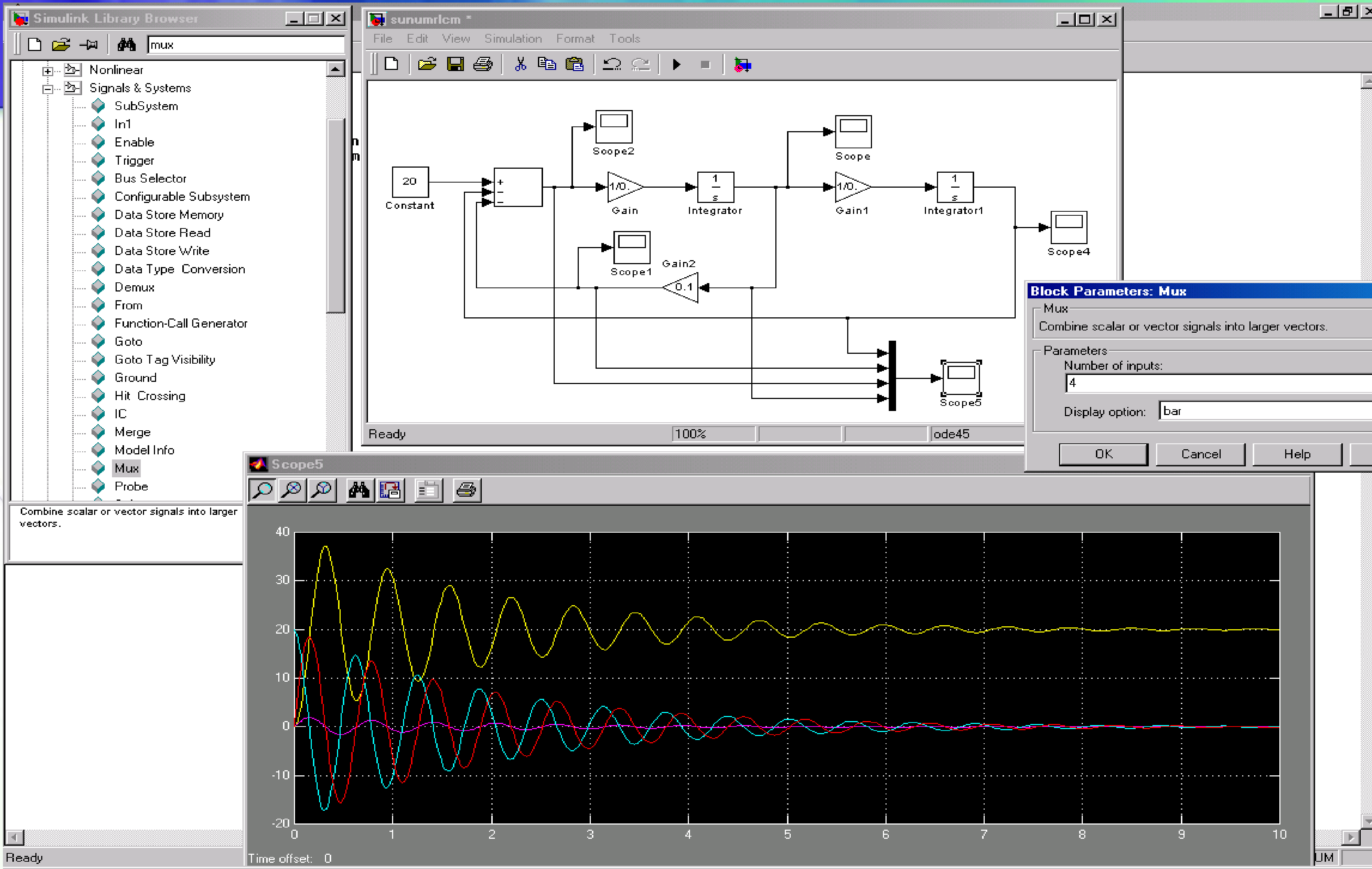
$$Vdc - Ri - Vc = L\frac{di}{dt}$$



$$i = C\frac{dVc}{dt} \qquad V_L = L\frac{di}{dt}$$

$$Vr = Ri$$



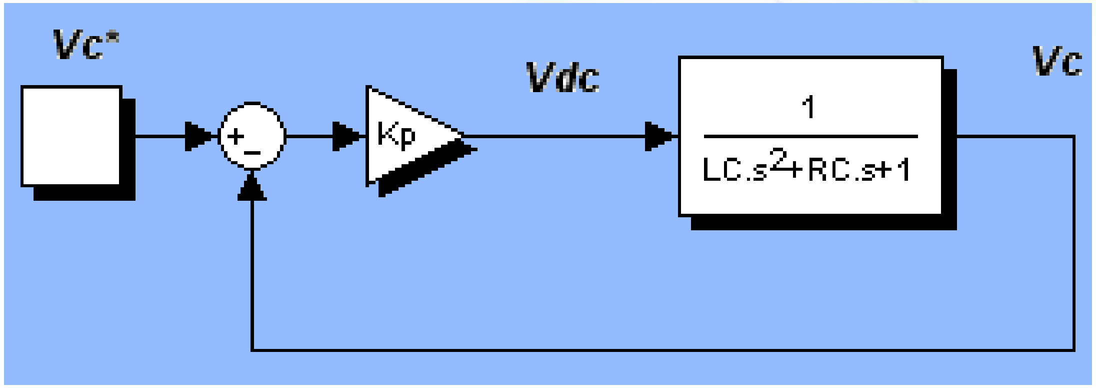


❖RLC DEVRESİNDE MUX KULLANILARAK SİMÜLASYON SONUÇLARININ ELDE EDİLMESİ

Matlab & Simulink

➢RLC DEVRESİ

(kapasite gerilimi kontrolü)



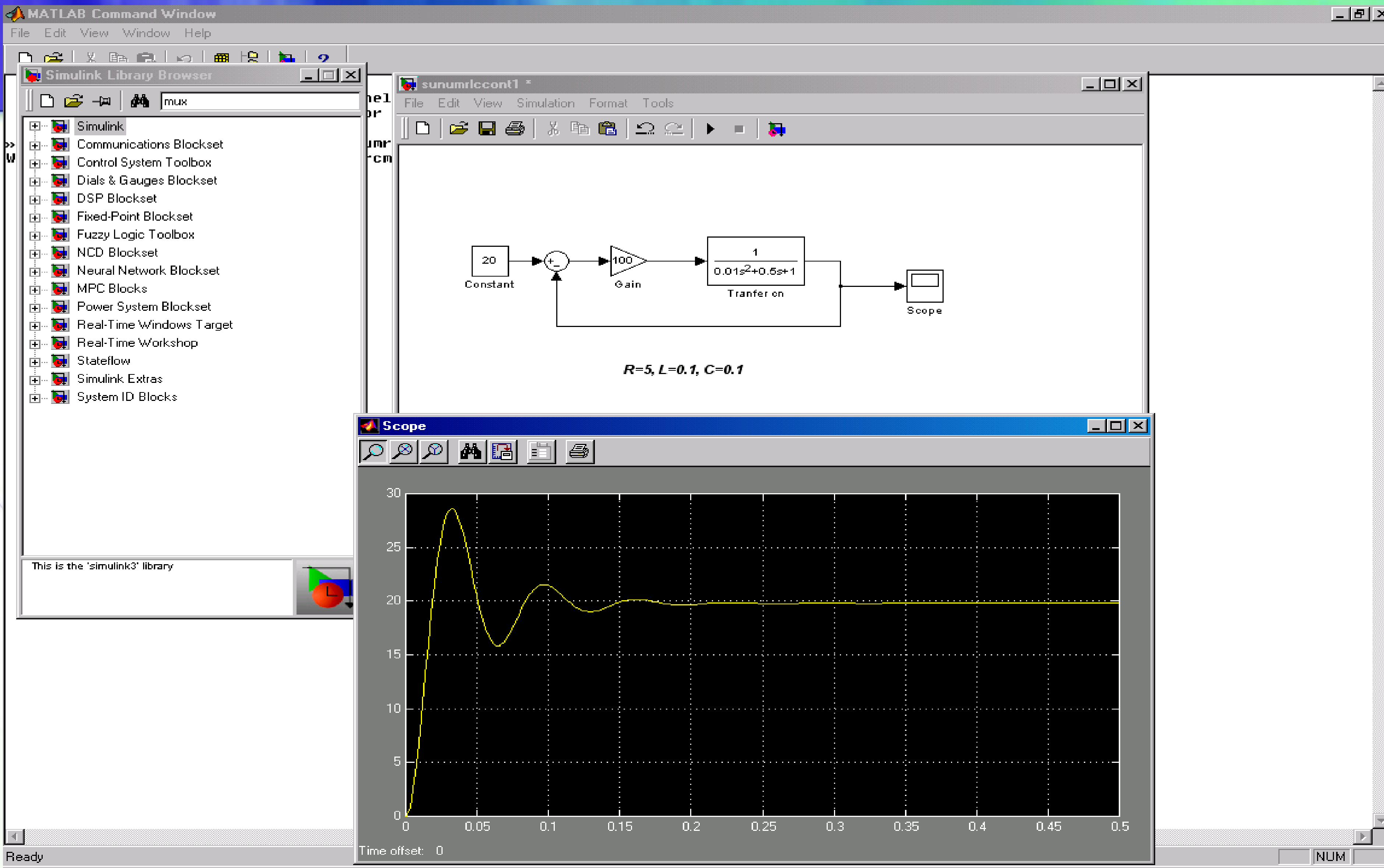
$$Vdc = Vr + Vc + VL$$

$$Vdc = RC\frac{dVC}{dt} + LC\frac{d^2Vc}{dt} + Vc$$

$$Vdc(s) = sRCVc(s) + s^2LCVc(s) + Vc(s)$$

$$Vc(s) = \frac{1}{s^2LC + sRC + 1}Vdc(s)$$

Matlab & Simulink



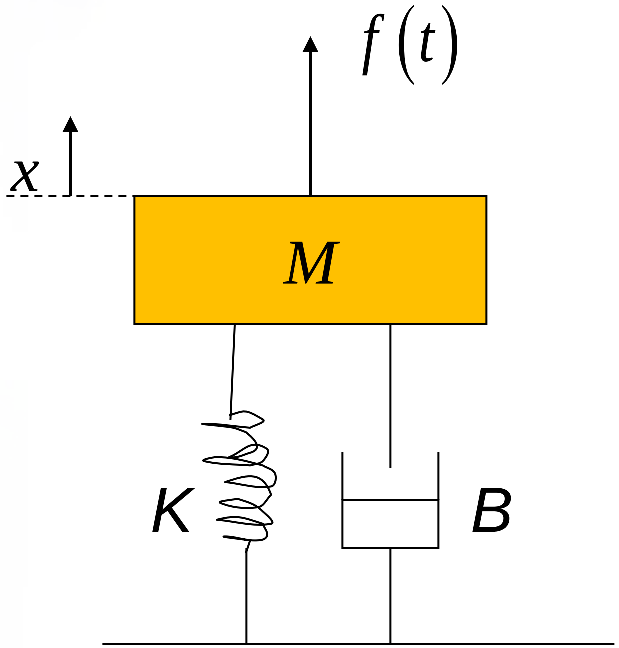
❖KAPASİTE GERİLİMİ KONTROLLÜ RLC DEVRESİ SİMÜLASYONU

Matlab & Simulink

Alt blok oluşturma

- **Subsystem – similar to “Subroutine”**
- **Advantage of Subsystems:**
  - **Reduce the number of blocks display on the main window (i.e. simplify the model)**
  - **Group related blocks together (i.e. More organized)**
  - **Can create a hierarchical block diagram (i.e. you can create subsystems within a subsystem)**
  - **Easy to check for mistakes and to explore different parameters**

Bir dinamik sistemin örneği: kütle-yay –sönüm sistemi



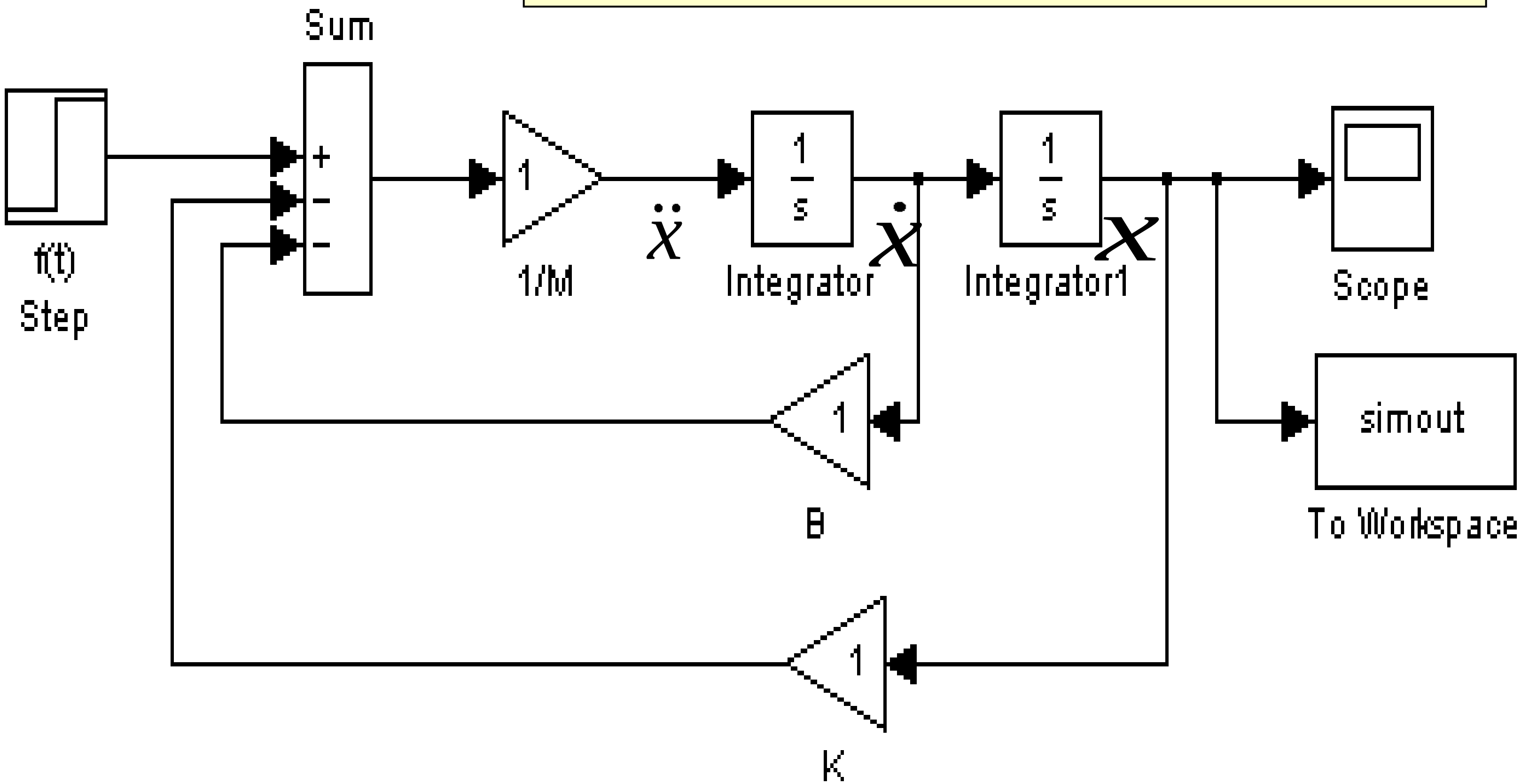
Sistemin matematiksel modeli:

$$\ddot{x} = \frac{1}{M}\left(-B\dot{x} - Kx + f(t)\right)$$

M=2kg; *B* = 2 Ns/m; *K*=2 N/m olsun

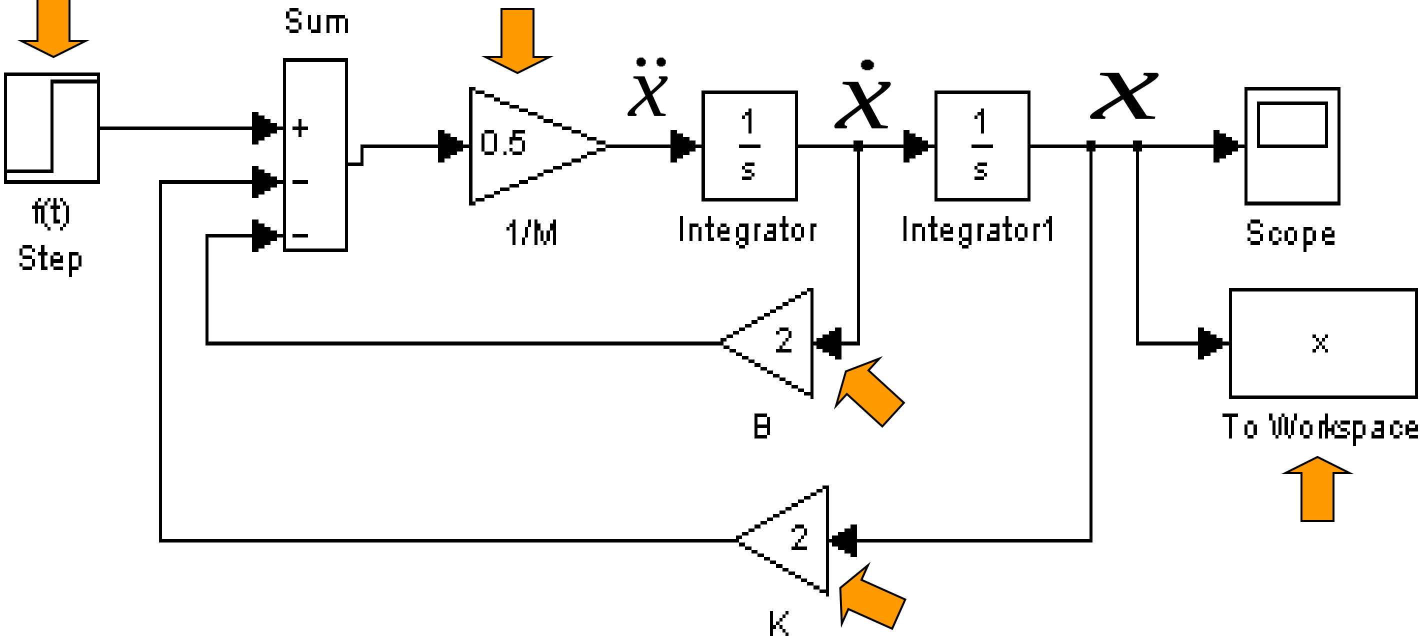
$$\boxed{\ddot{x} = \frac{1}{2}\left(-2\dot{x} - 2x + f(t)\right)}$$

$$\ddot{x} = \frac{1}{2}\left(-2\dot{x} - 2x + f\left(t\right)\right)$$



$$\ddot{x} = \frac{1}{2}\left(-2\dot{x} - 2x + f(t)\right)$$

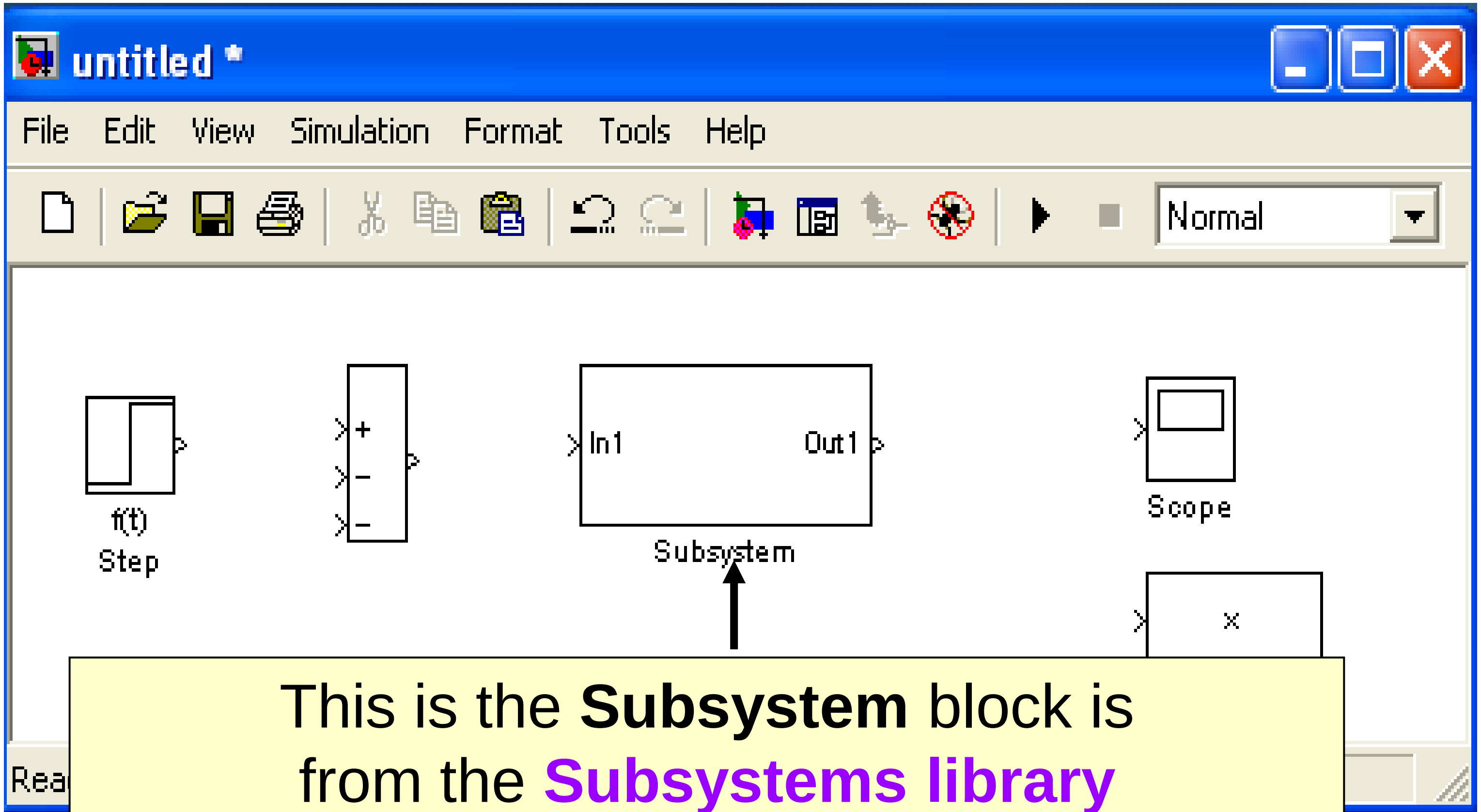
Step time =0 olarak ayarla



Not: Tüm başlangıç şartlarını= al.

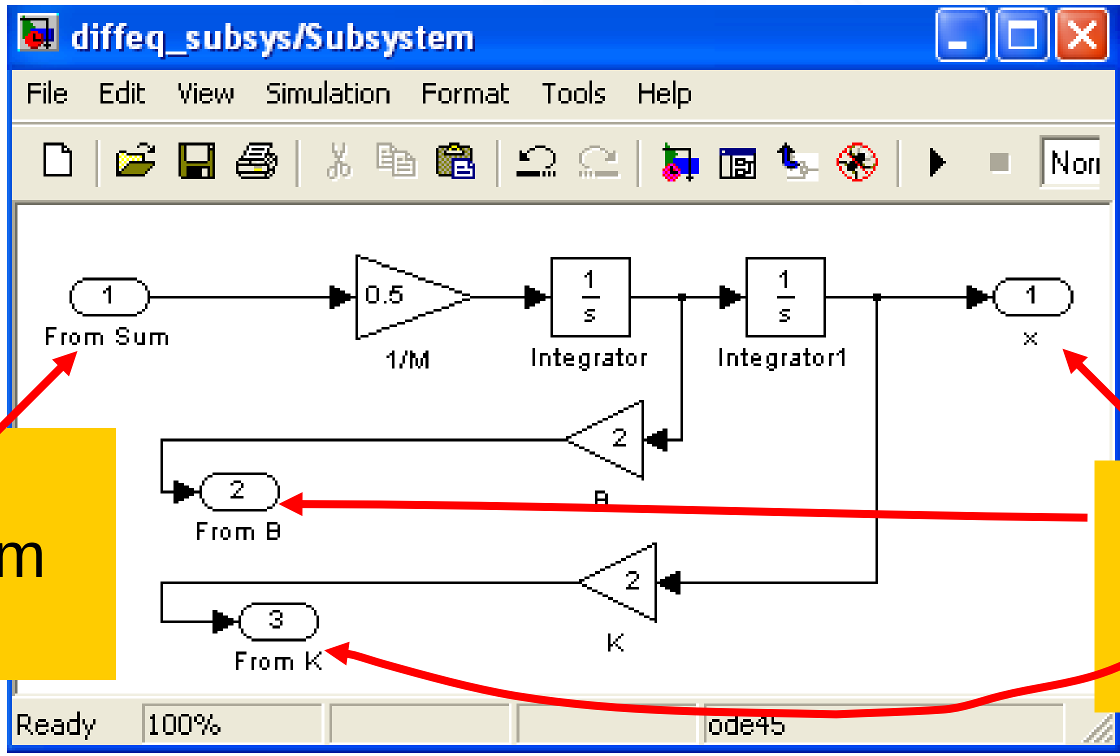
Create Subsystem

STEP 1: Creating Blocks (Main window)



This is the **Subsystem** block is from the **Subsystems library**

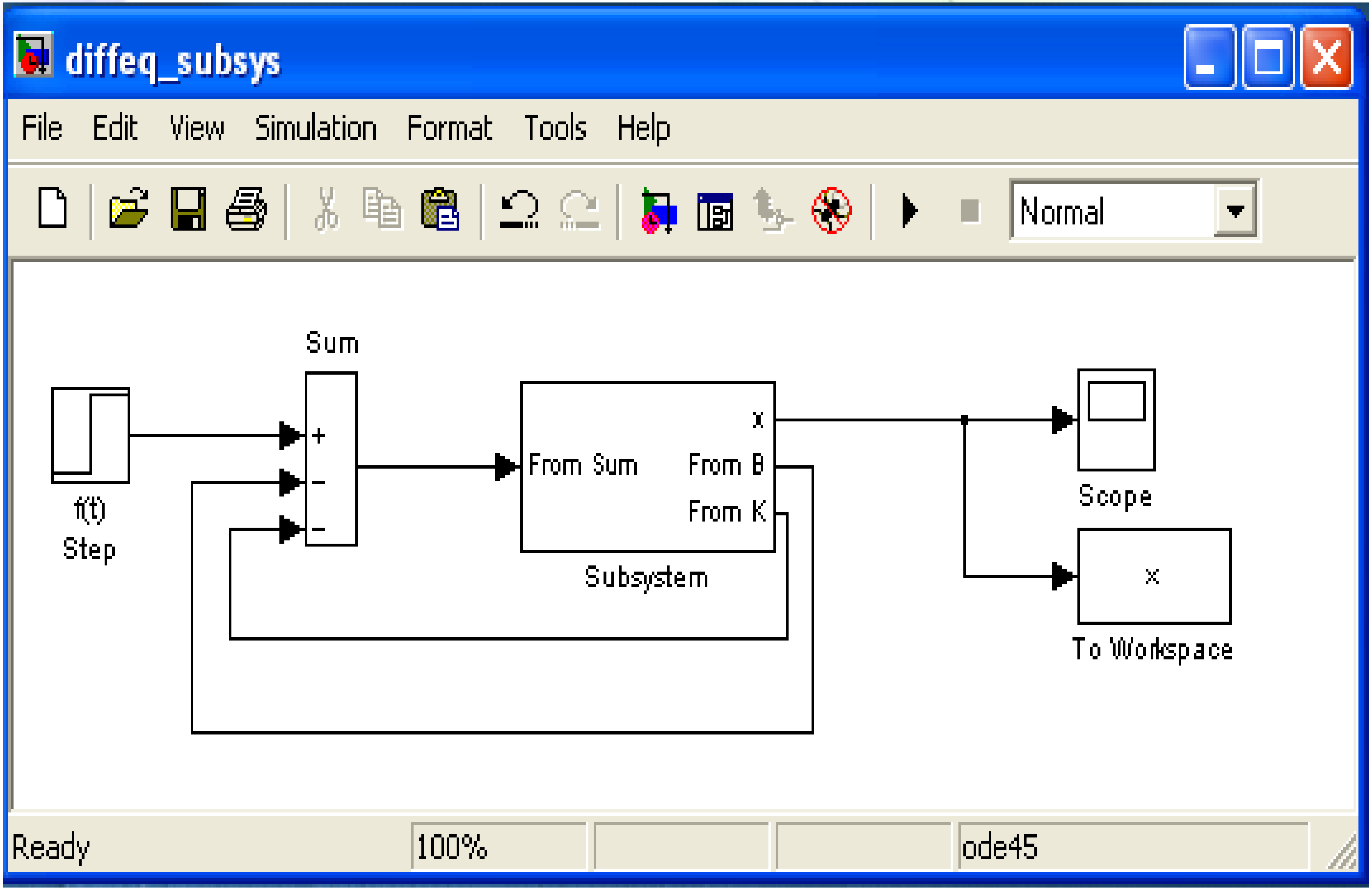
STEP 2: Double click Subsystem block and create a *model* in the Subsystem block



Inport (named from "sum"

Outport (three outports)

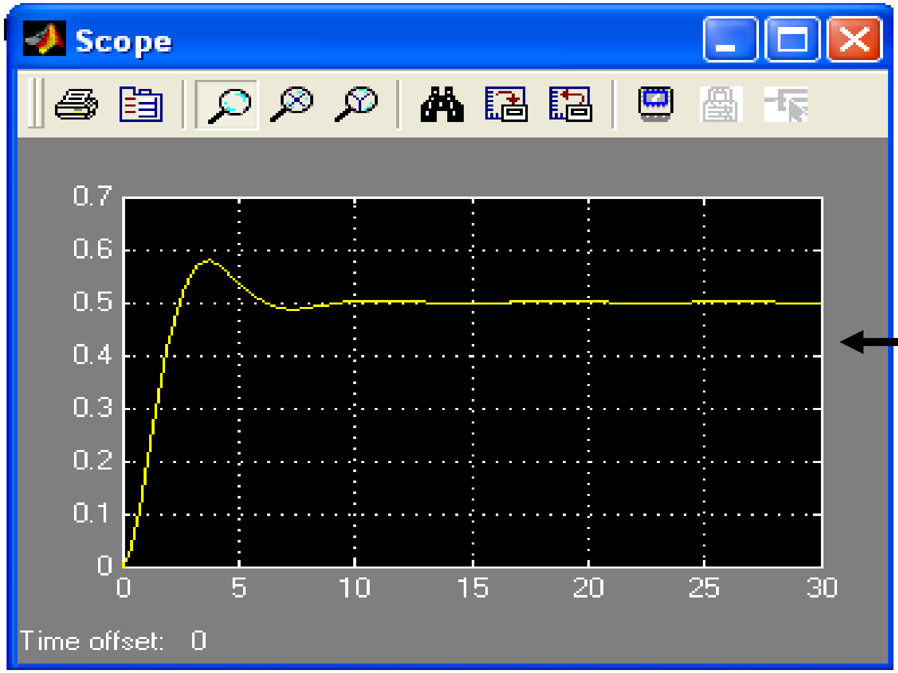
STEP 3: Making connections (Main window)



STEP 4: Set Parameter (Main window)

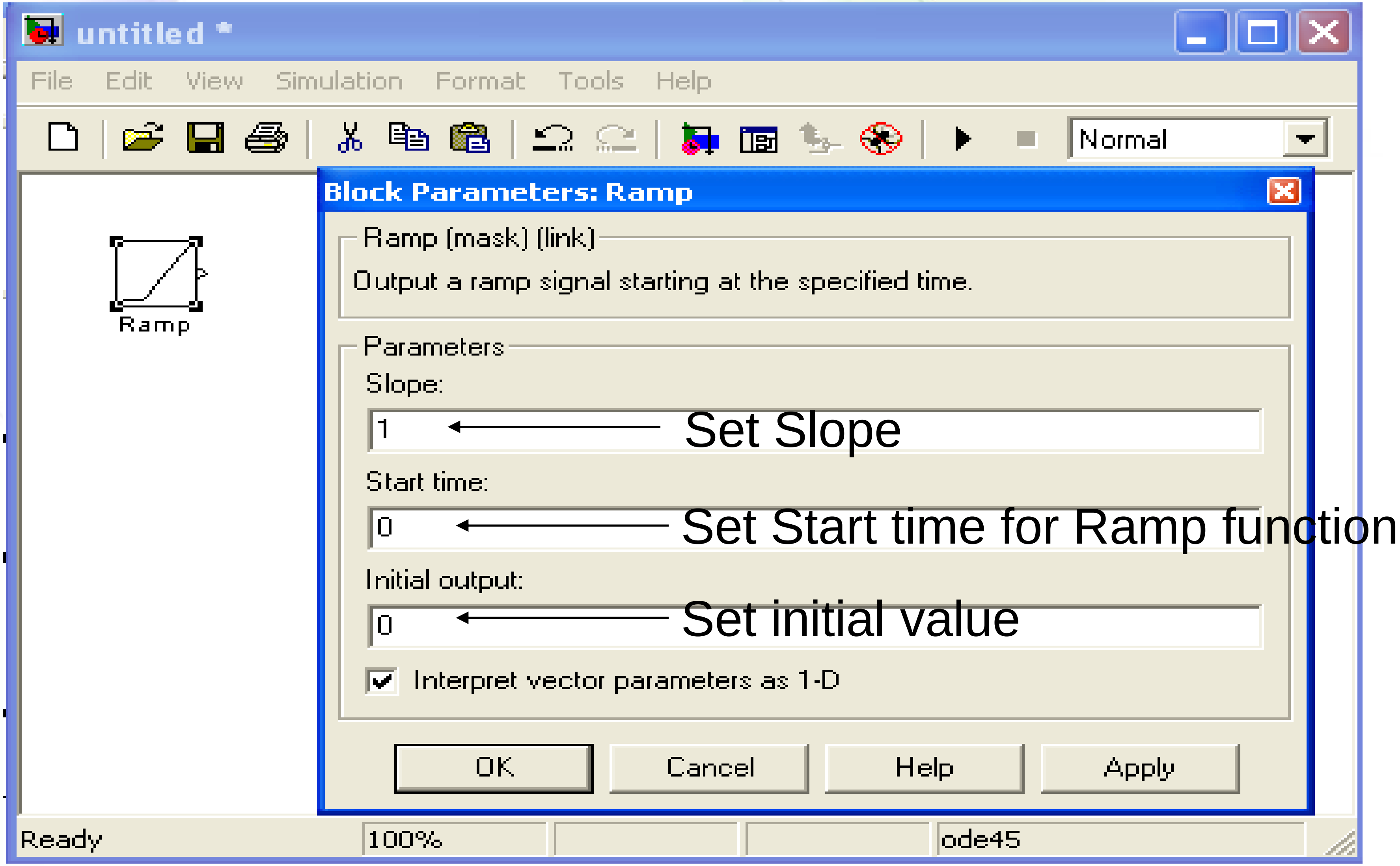
STEP 5: Running Simulation

Then view output response

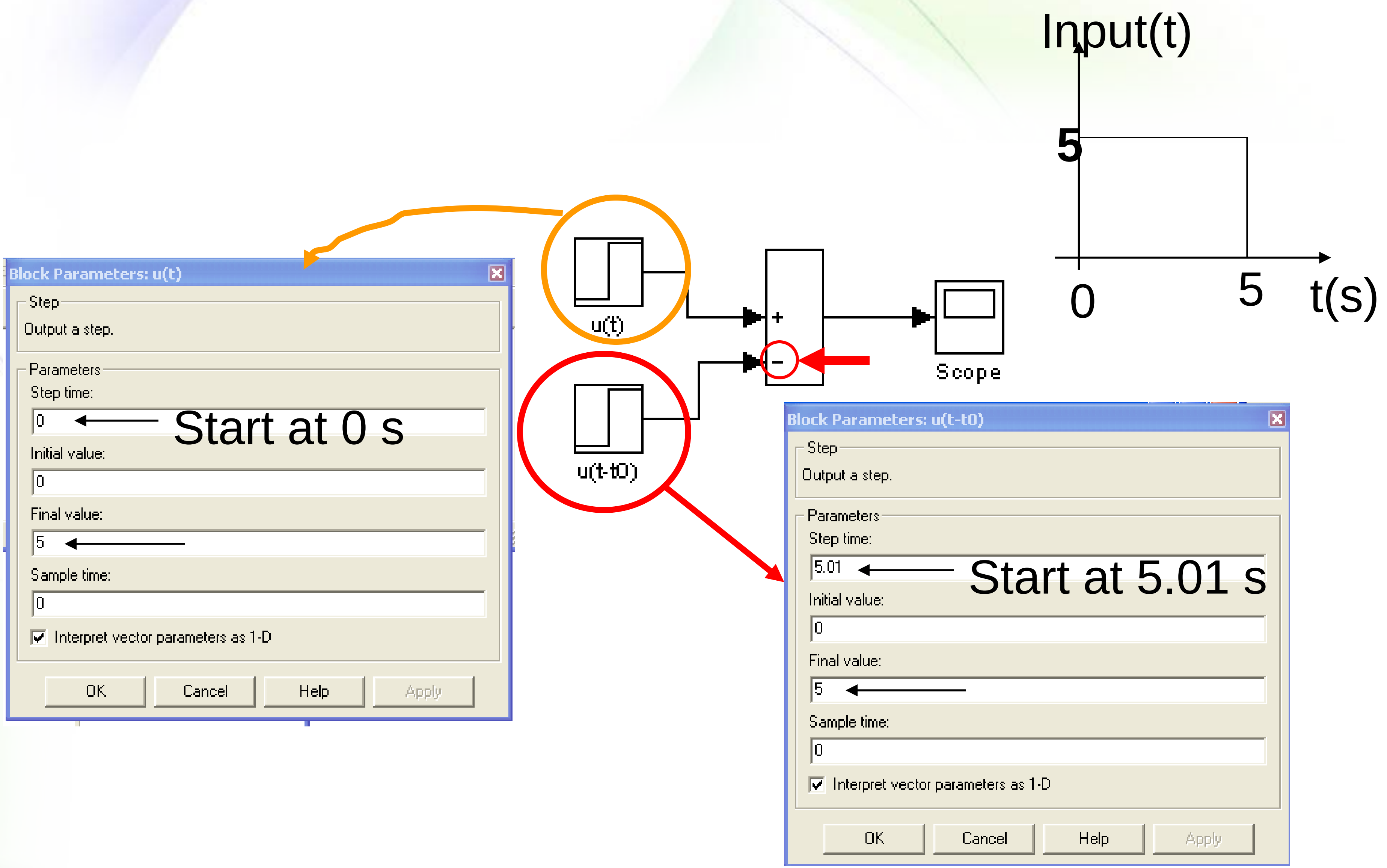


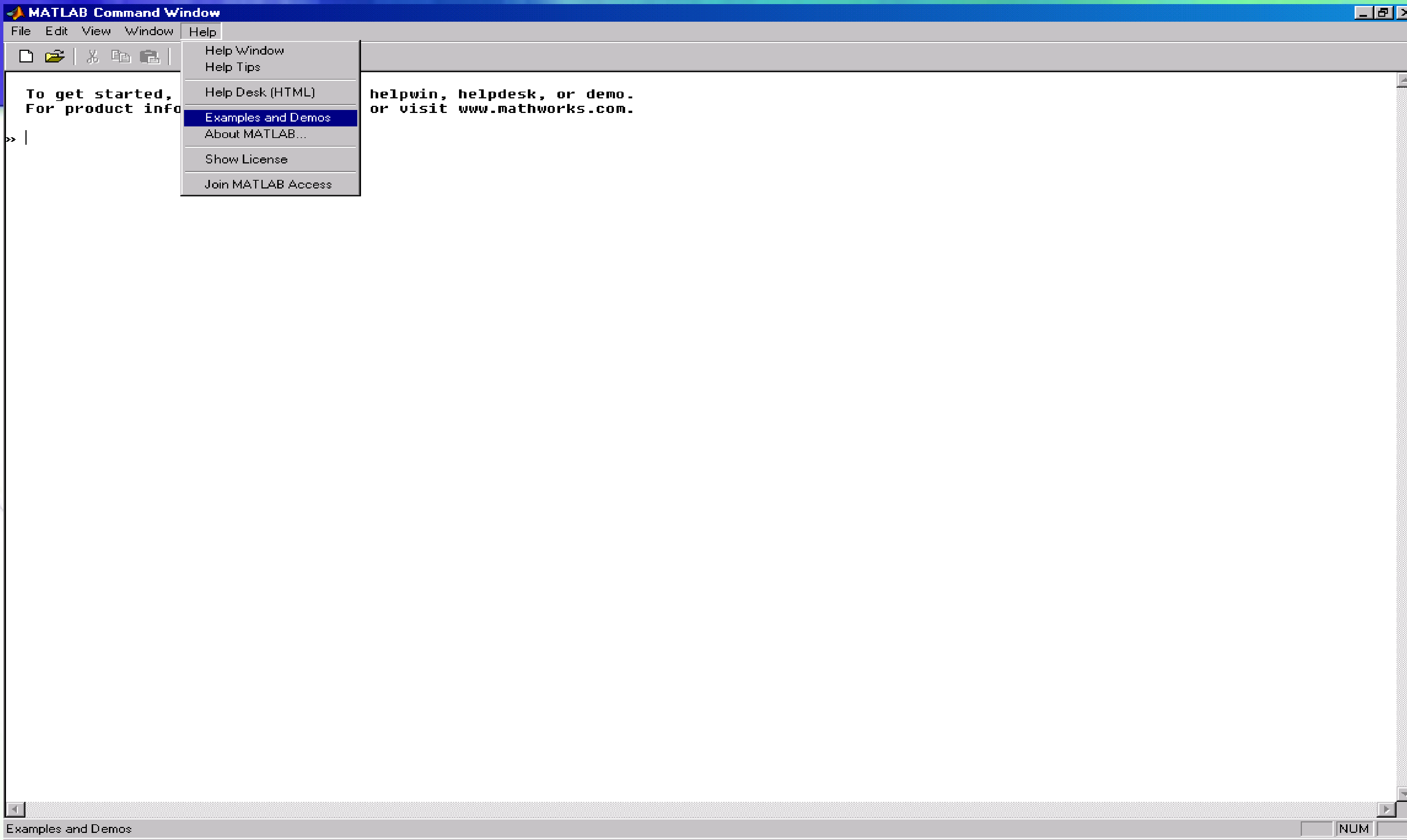
Output from Scope block

Ramp Function

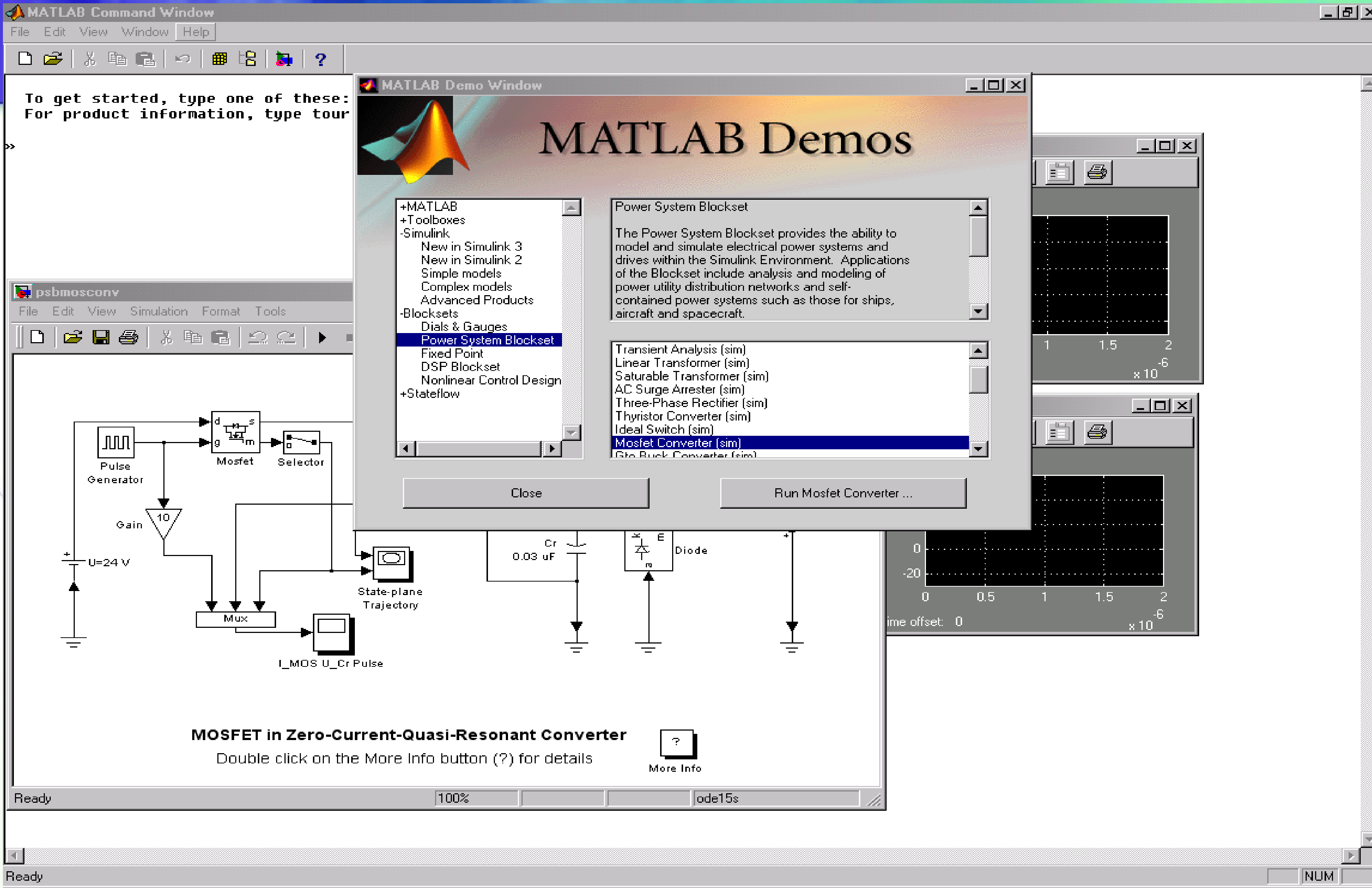


Unit Step Function or Impulse





❖SİMULİNK DEMOLARI



❖DEMO EKRANI

Matlab & Simulink