

# HPFFBU'15

## LabVIEW ve Arduino Tanıtım

+

## Uygulamalar



Görkem Türemen

01.02.2015



# İçerik

- LabVIEW
- Arduino
- Motorlar
- Uygulama



# Terimler ve Kısaltmalar

- **LabVIEW**: Laboratory Virtual Instrument Engineering Workbench
- **NI**: National Instruments
- **VI**: Virtual Instrument
- **GPIB (IEEE-488)**: General Purpose Interface Bus
- **LAN**: Local Area Network
- **PCI**: Peripheral Component Interconnect
- **VISA**: The Virtual Instrument Software Architecture
- **IVI**: Interchangeable Virtual Instrumentation



# Sözlük

- Data flow programming: Veri akışı programlama
- Virtual instrument: Sanal donanım
- Front panel: Ön pencere
- Block diagram: Akış çizelgesi
- Control: Yönlendirme
- Indicator: Gösterge
- Palette: Tahta
- Terminal: Uçbirim
- Wire: Tel
- Structure: Yapı
- Array: Dizi
- Cluster: Küme
- Shift register: Kaydırma yazgacı
- Graph: Çizim
- Express: Çabuk
- Node: Düğüm
- String: Dizgi
- Dynamic: Devinimsel



# LabVIEW

- Görsel bir programlama dili/ortamı/yazılımı'dır.
- Herhangi bir donanımın kontrolünün bir akış şizelgesi içinde bilgisayar destekli olarak yapılmasını sağlar.
  - Fiziksel donanım → Sanal donanım (VI)
- İlk olarak 1986'da Apple'ın MAC'i için geliştirildi.
- Kullandığı görsel programlama diline "G" deniliyor.
- Mac OS X, Linux/UNIX ve MS Windows'da çalışabiliyor.
- Ücret: 2.400 - 13.100 TL arasında değişiyor.



# LabVIEW

## -Kullanım-

- Üç ana iş için kullanılabilir:
  - Veri toplama
  - Veri işleme (çözümleme)
  - Donanım yönlendirme

Donanım →  
Sıcaklık algıci



→ Donanım  
İklimlendirici



# LabVIEW

## -Dil farkı-

- Görsel programlama yöntemi sayesinde C/C++, Fortran, vb. programlama dillerindeki komutlar yerine simgeler ile program yazılır.
- Metin tabanlı dillere kıyasla çok daha kısa sürede programlama yapılabilir.

```

float celsius(float fahrenheit);

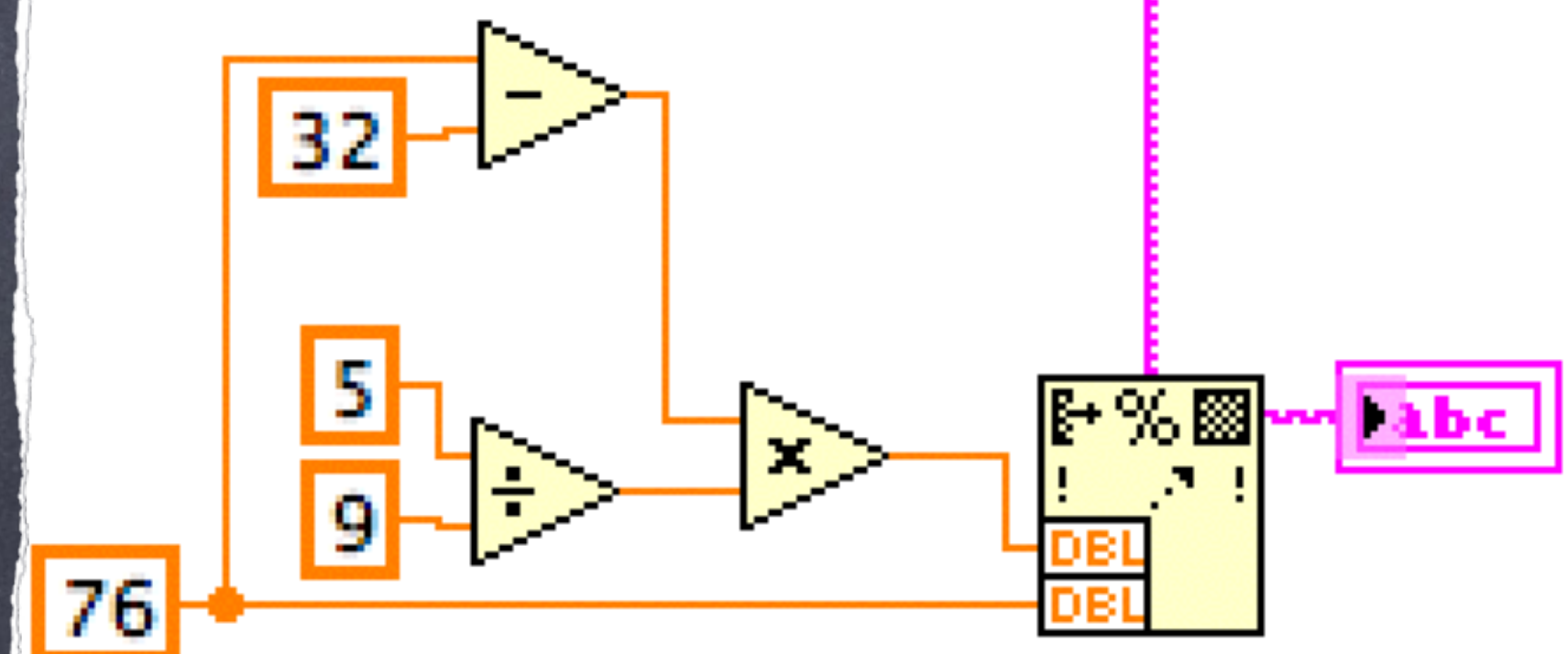
main()
{
    float DegreeF=76;
    float DegreeC=celsius(DegreeF);
    printf("The Temperature is %f F and %f C\n", DegreeF, DegreeC);
}

float celsius(float fahrenheit)
{
    float celsius = (5.0/9.0)*(fahrenheit-32);
    return celsius;
}

```

C

G The Temperature is %f F and %f C



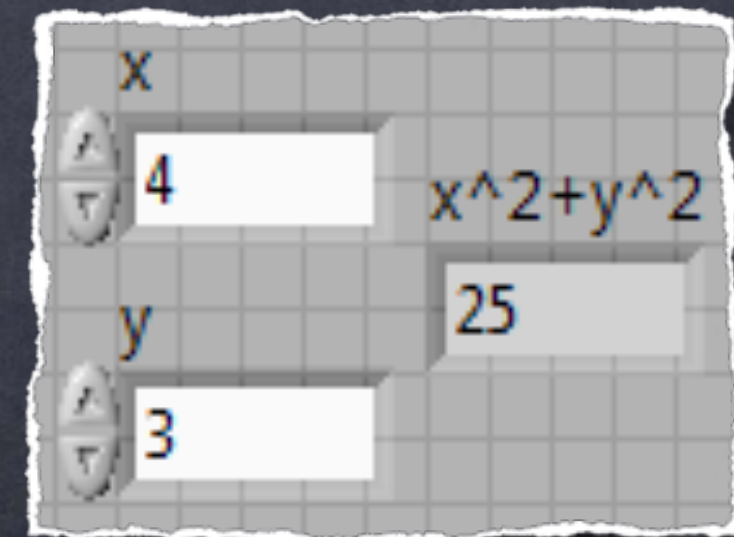
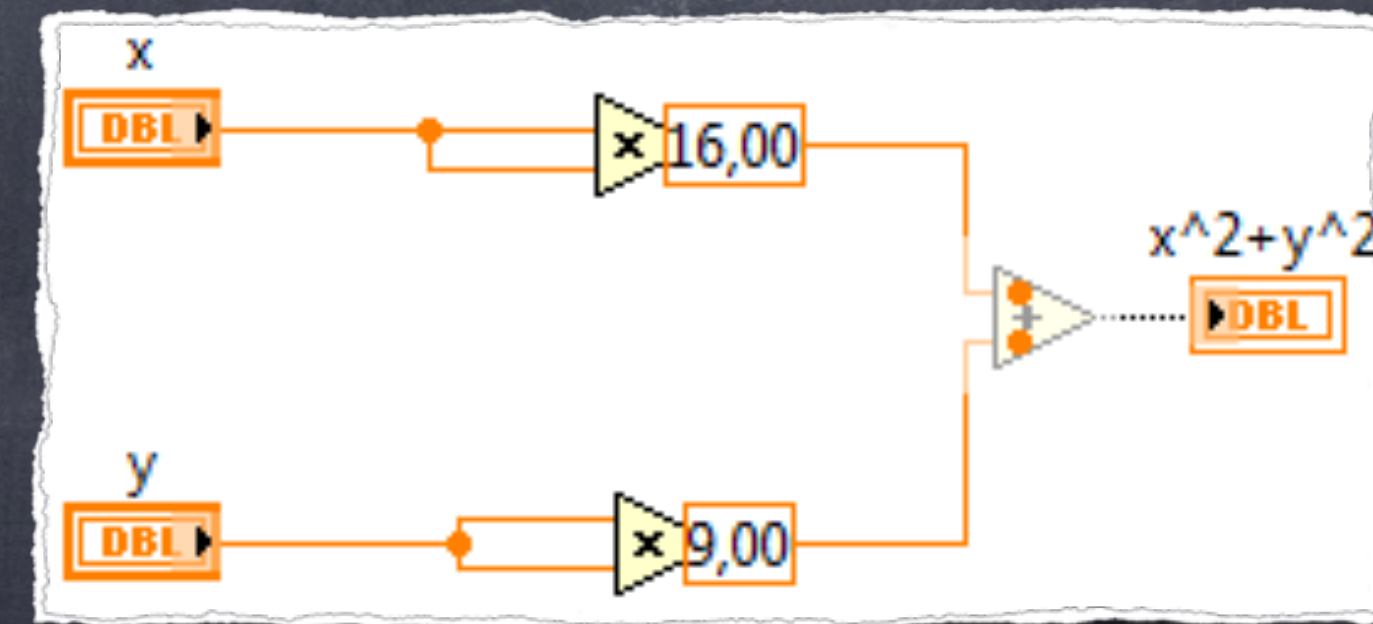
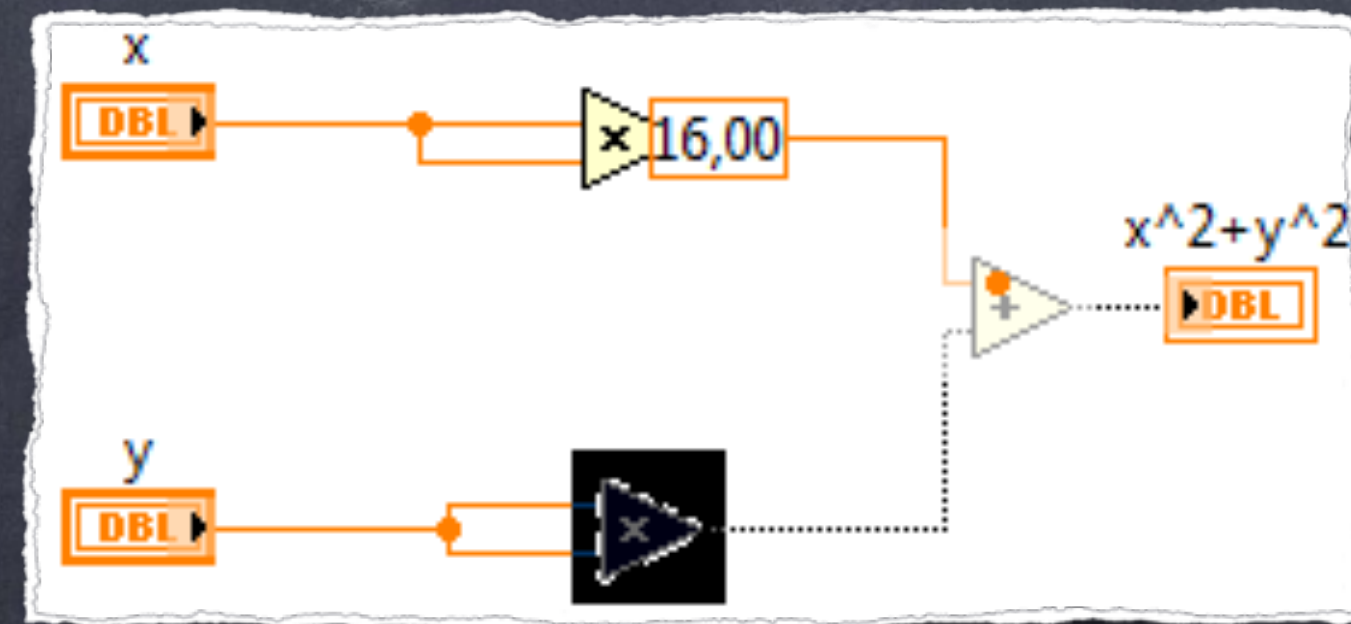
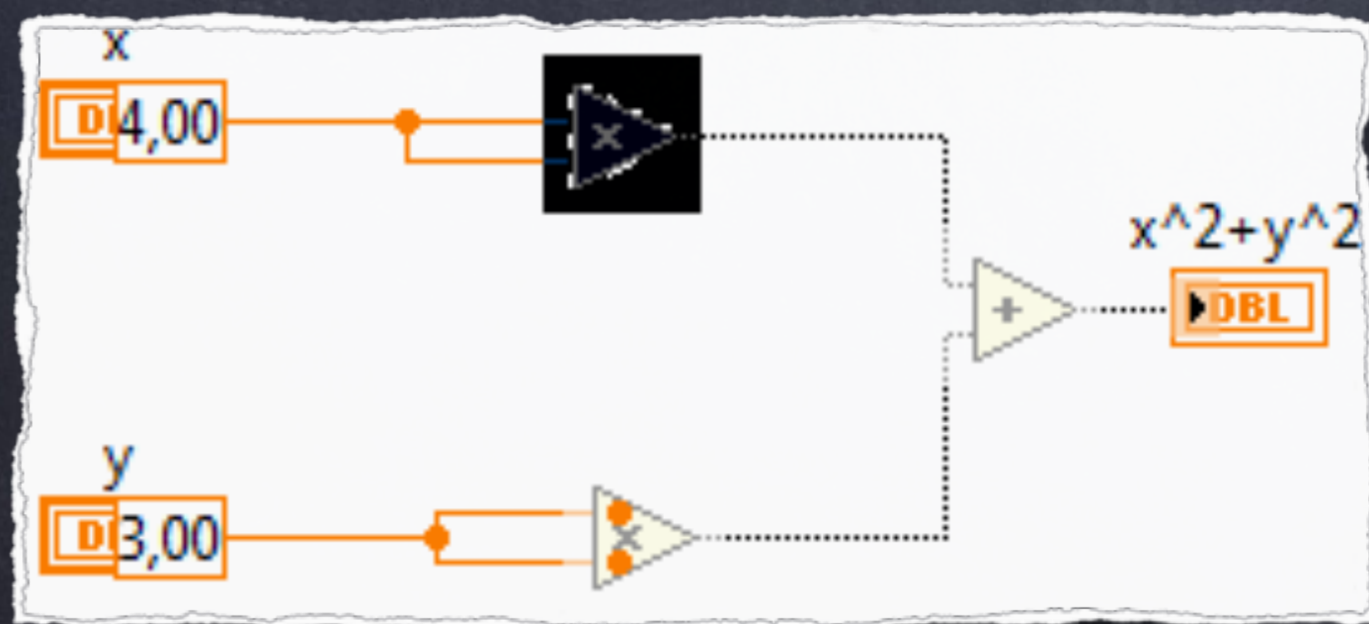
The Temperature is 24.444444 F and 76.000000 C



# LabVIEW

## -Veri akışı programlama-

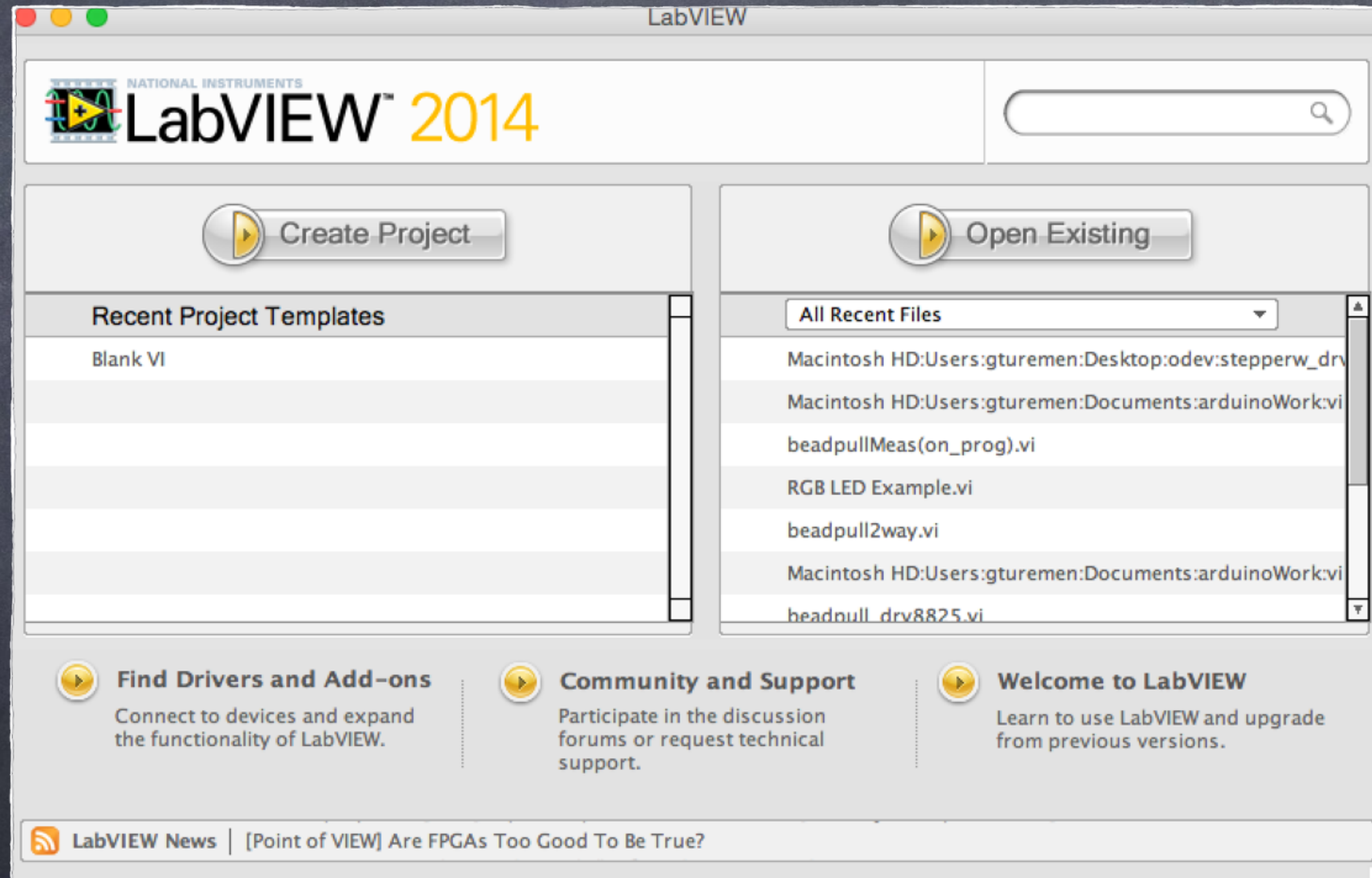
- Programların çalışma sırasını akış çizelgesi üzerindeki yollardan verinin akma sırası belirler.
- Akış çizelgesindeki herhangi bir bileşenin geliştirilmesi için, gereken tüm girdi verilerinin o bileşene ulaşmış olması şarttır.
- Bu sayede çoklu işlemler eşzamanlı olarak gerçekleştirilebilir.





# LabVIEW

-Program çevresi-

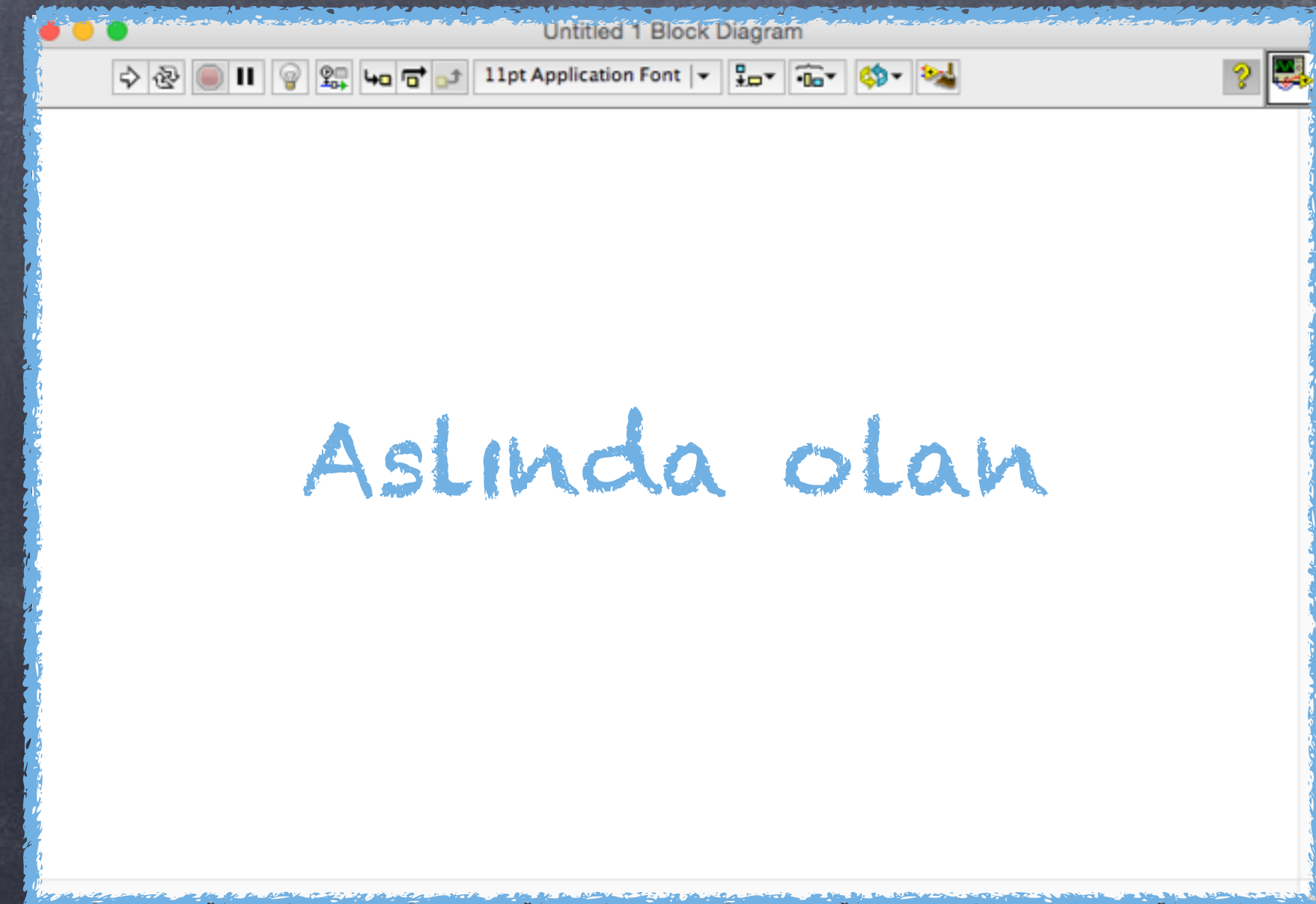




# Program evresi

## -VI-

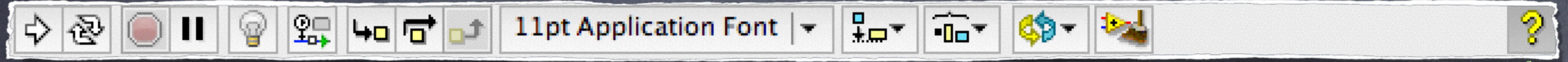
- LabVIEW programlarının her birine VI denir.
- n pencere ve akış izelgesi olmak üzere iki kısımdan oluşur.
- Her bir programın kütük uzantısı → \*.vi





# VI

## -Araç çubuğu-



Hata ayıklama

Hizalama

İçerik yardımı



VI'nın geliştirilmeye uygun olduğunu gösterir ve tıklandığında çalışmasını sağlar.



VI'nın çalışmakta olduğunu gösterir.



VI'da bir hata olduğunu ve geliştirilmeye uygun olmadığını gösterir.



VI'nın durdurulmasını sağlar. Son çare olarak kullanılır !!!



VI'nın duraklatılmasını sağlar. Duraklatılan konumdan devam edilebilir.



VI'ın durdur yada duraklata tıklanmadıkça tekrar tekrar çalışmasını sağlar.



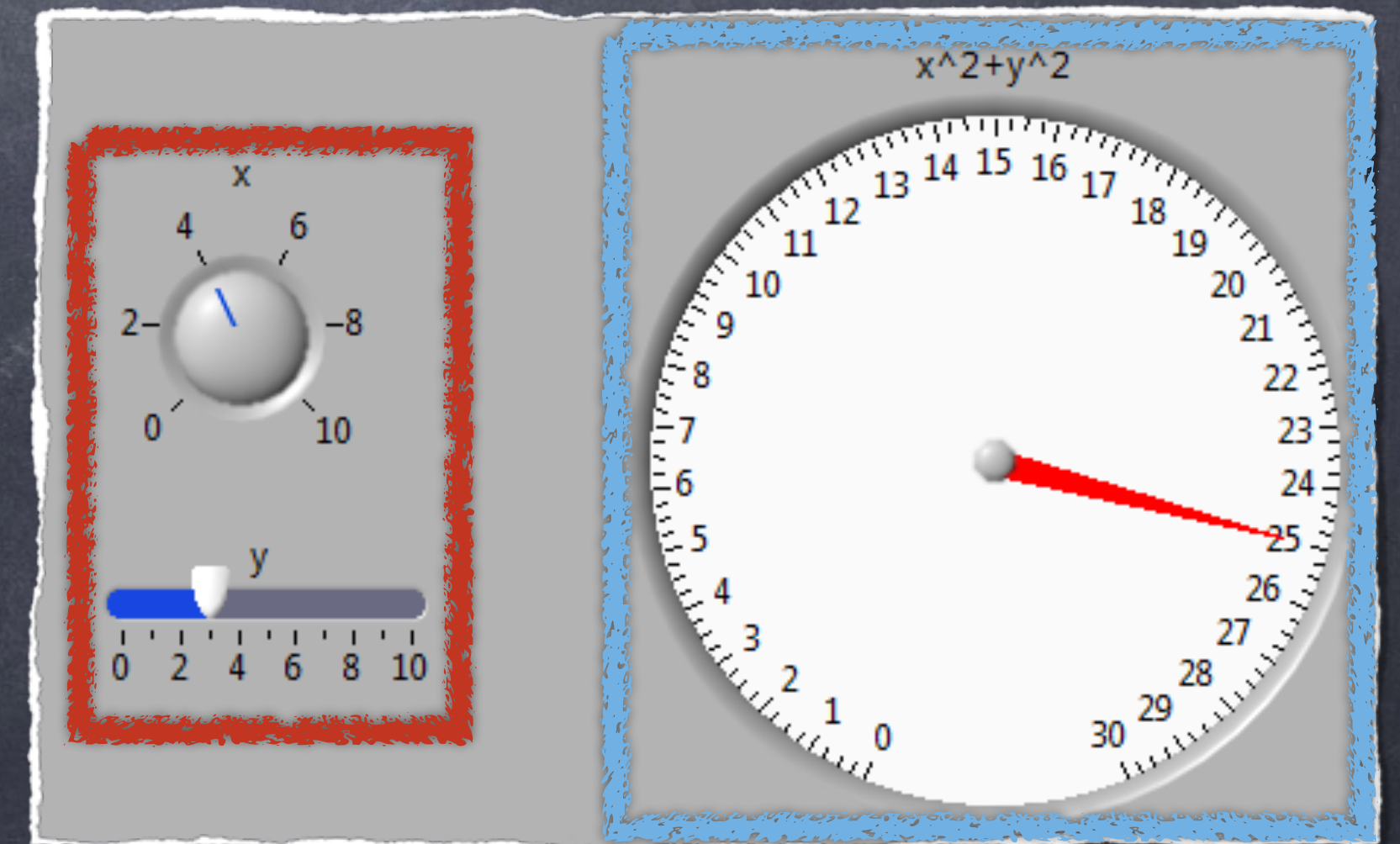
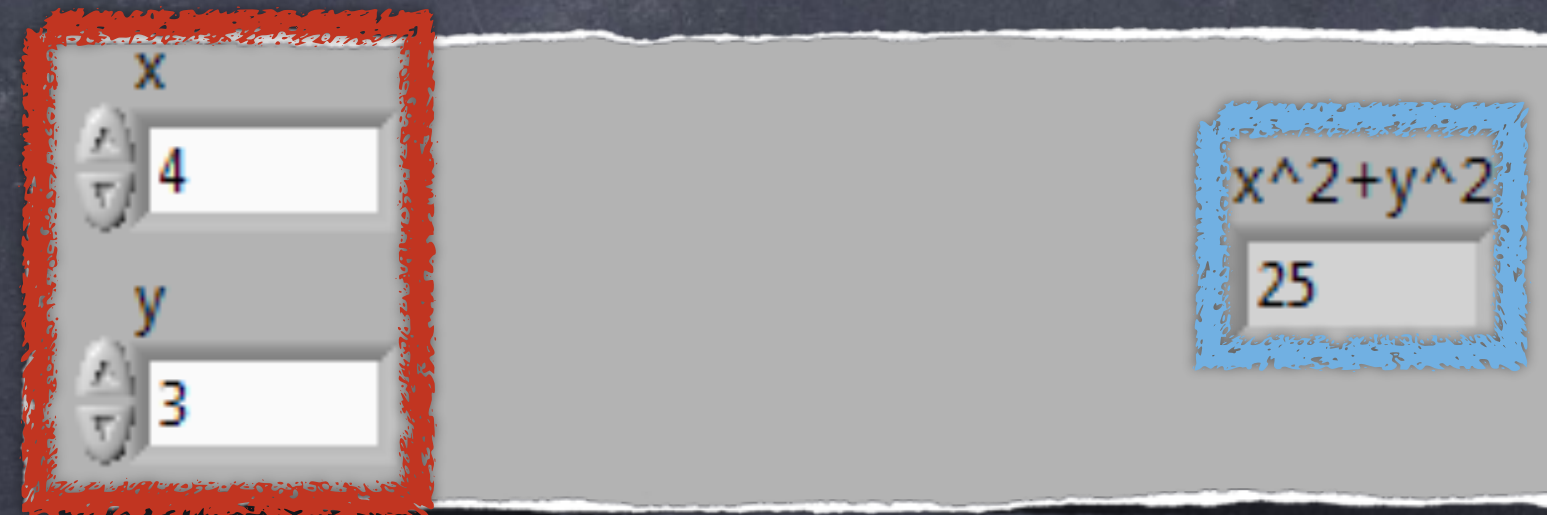
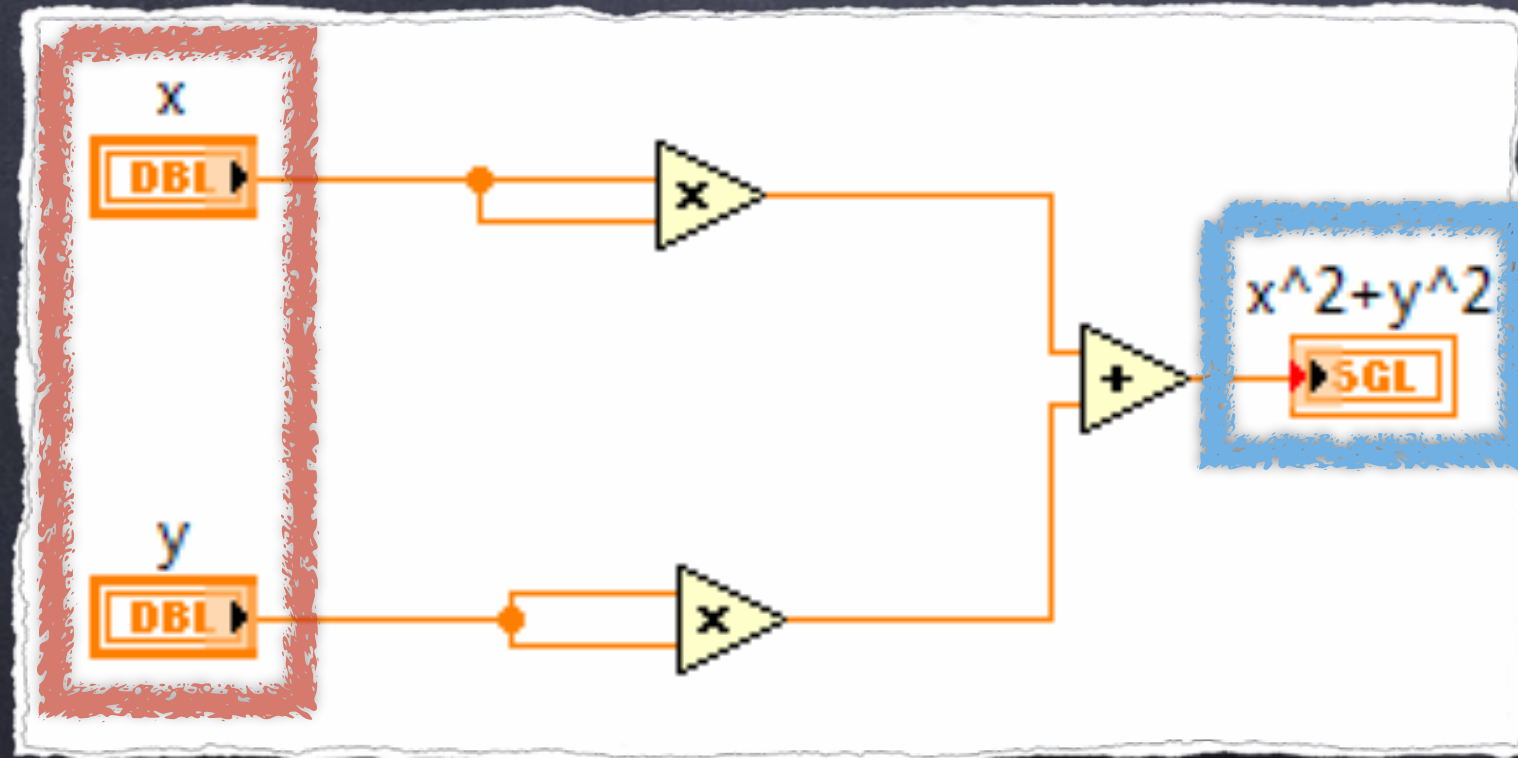
VI'nın çalışması sırasında verinin akışını hareketli olarak gösterir.  
(Sadece akış çizelgesinde kullanılabilir.)



# Programlama Ortamı

## -Ön pencere-

- Ön pencere, akış çizelgesinde oluşturulan VI'nın kullanıcı arayüzlerini içerir.
- **Yönlendirme:** VI'ya verilecek girdileri yönlendirmeye/ayarlamaya yarar. Ör: çevirmeli düğme, basmalı düğme, kadran, vb...
- **Göstergeler:** VI'dan alınacak çıktıları görüntülemeye/yazdırmaya yarar. Ör: çizim, LED gösterge, sayaç, vb...

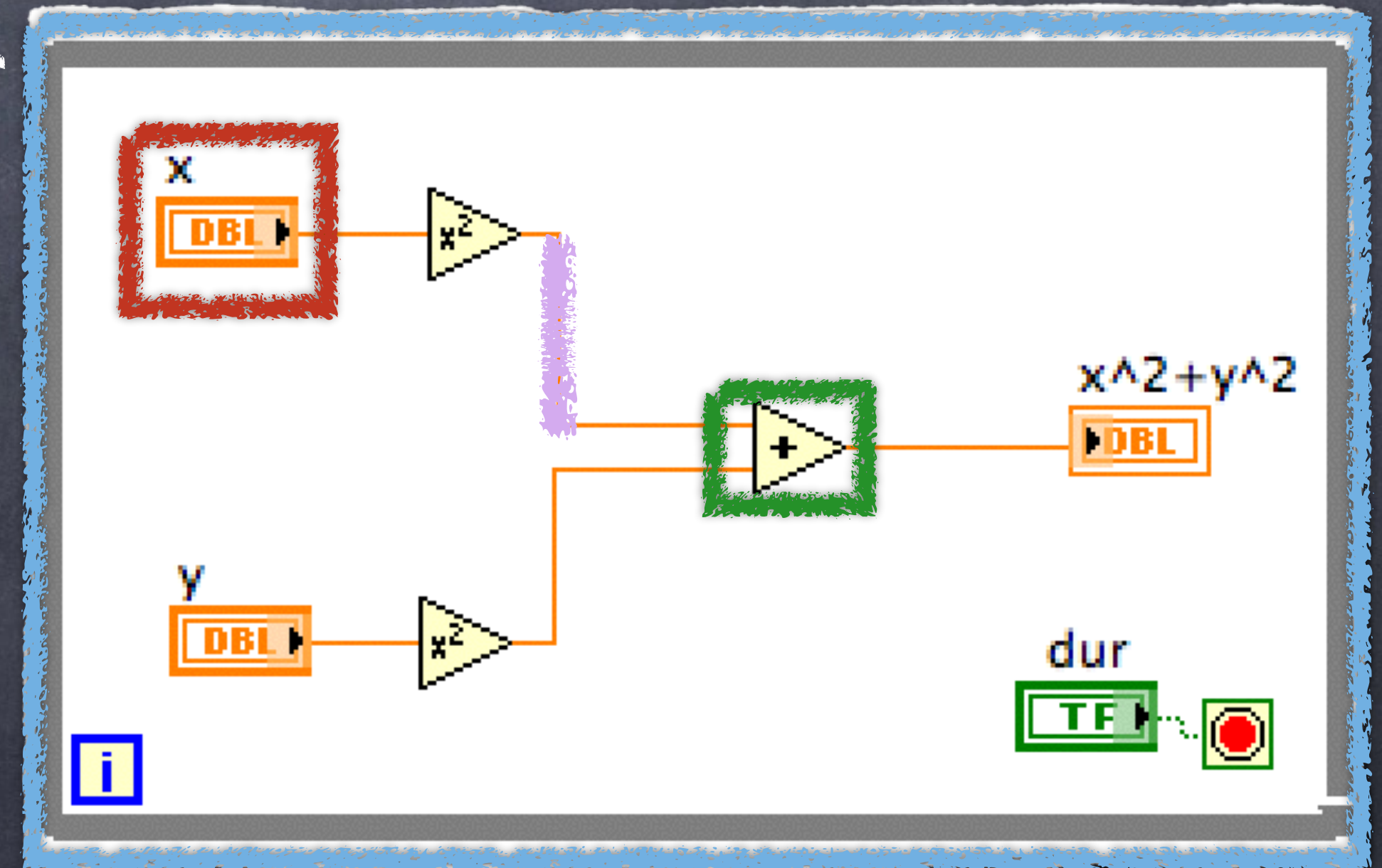
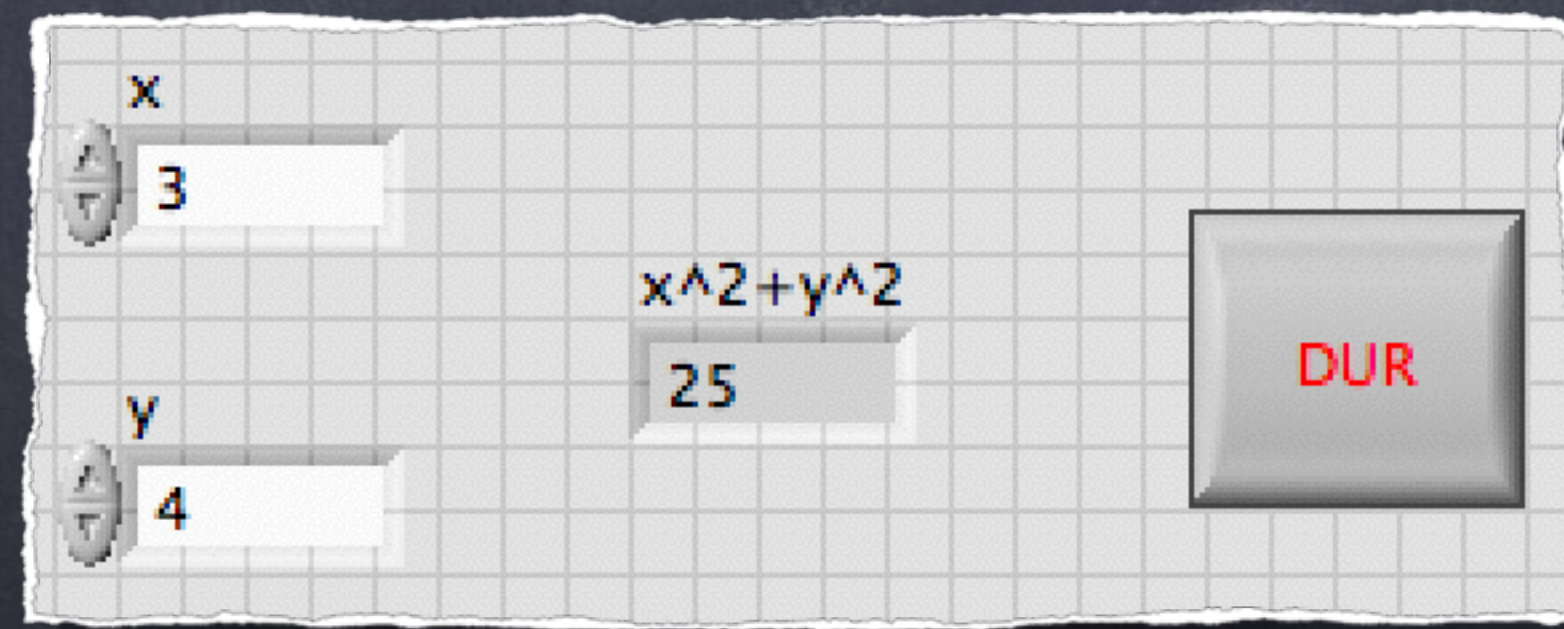




# Programlama Ortamı

## -Akış Çizelgesi-

- Akış çizelgesi ön pencerenin altında yatan program yapısını içerir ve dört ana bileşenden oluşur:
- **Uçbirim Simgeleri:** Ön penceredeki yönlendirme ve göstergelere doğrudan bağlıdır. Ön pencere ile akış çizelgesi arasındaki veri alışverişini sağlarlar.
- **Düğümler:** Belirli bir işlevi yapmak üzere girdi ve çıktı bağlantı noktaları olan nesnelerdir.
- **Yapılar:** Koşullar ve döngüler ile VI'nın akışını belirlerler.
- **Teller:** Akış çizelgesinde verinin akmasını sağlarlar.

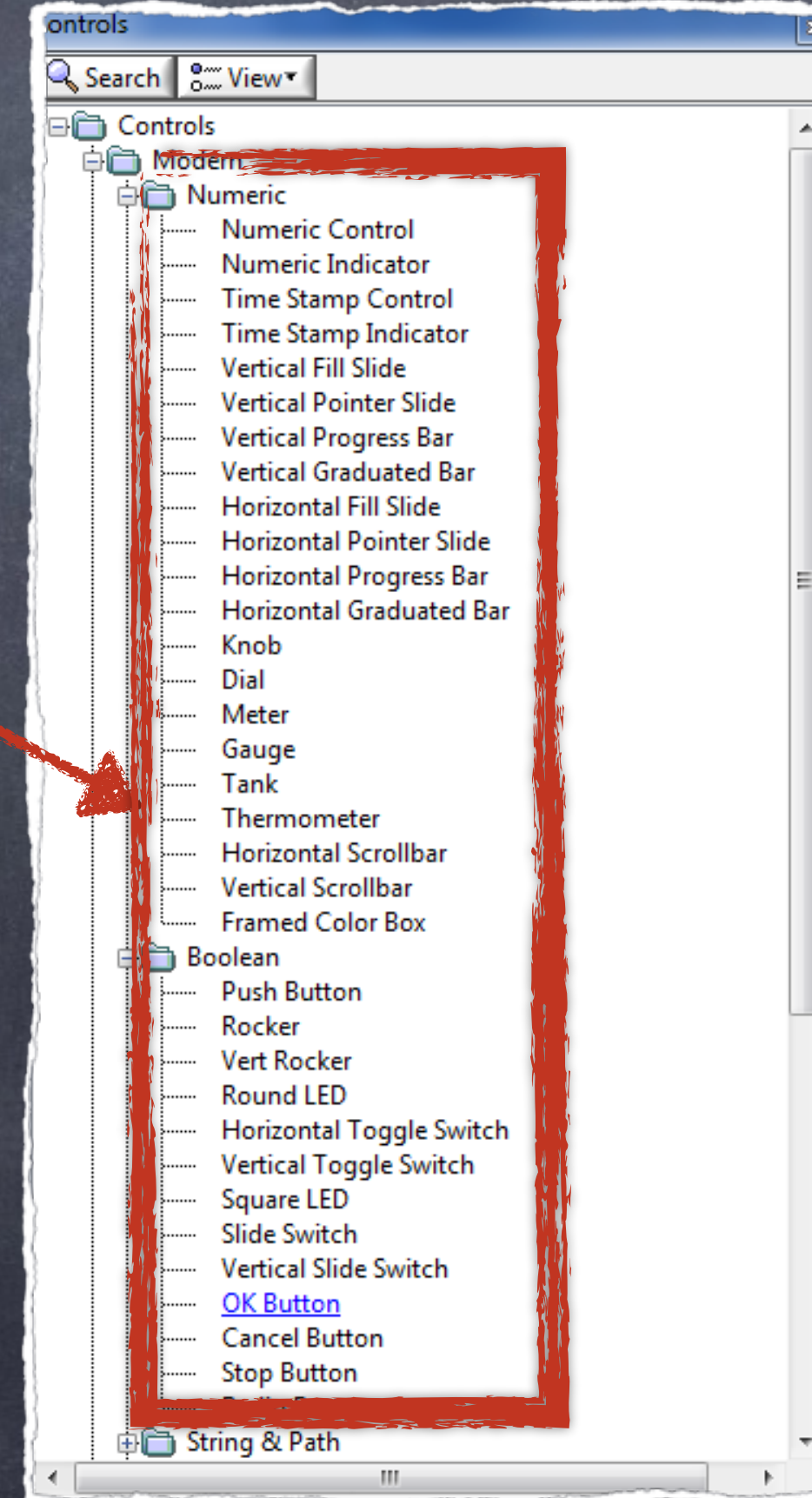
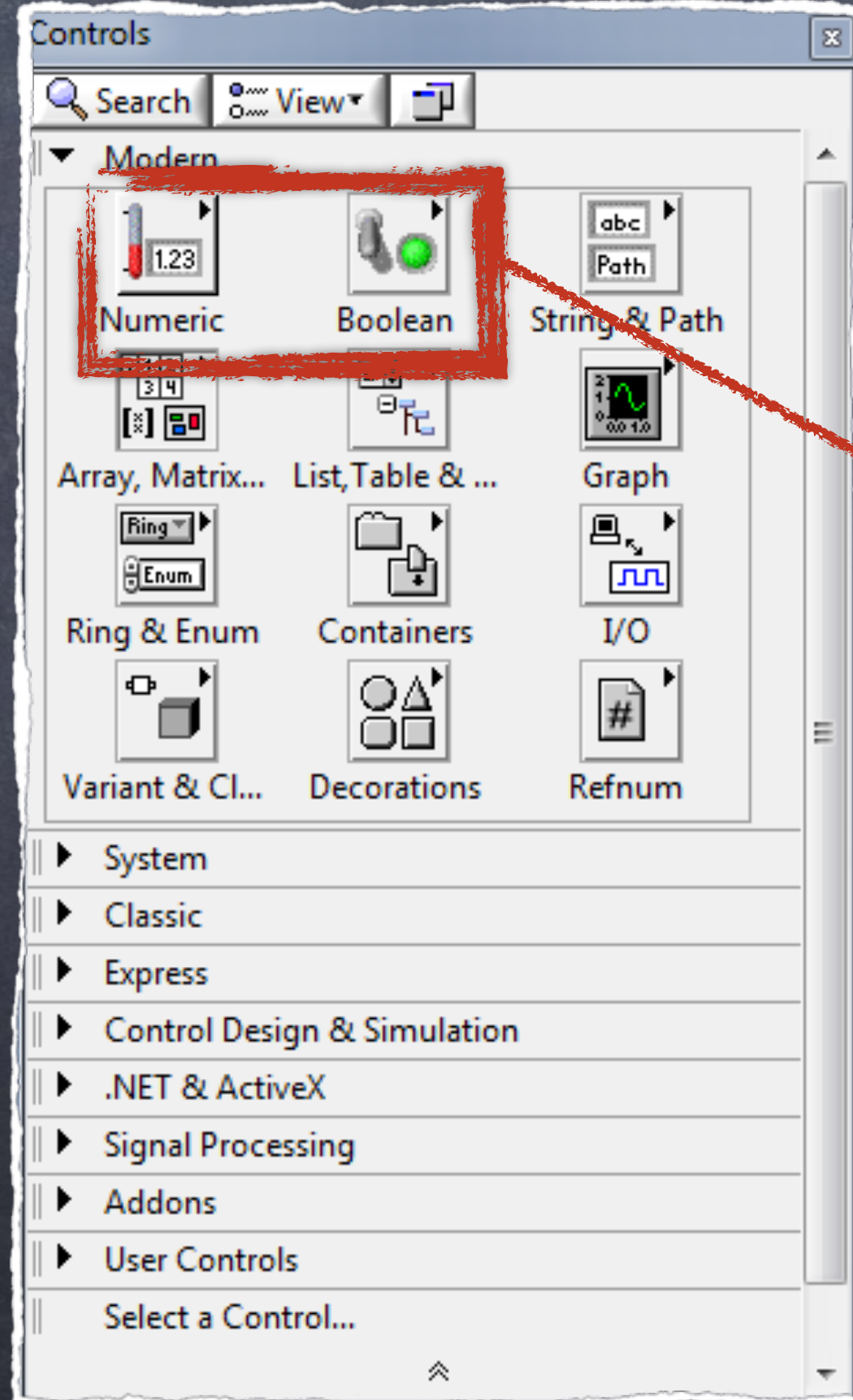




# Tahtalar

## -Yönlendirme Tahtası-

- VI'da kullanılacak yönlendirme ve göstergeleri barındırır.
- Ön pencere üzerinde sağ tıklanarak ulaşılabilir.

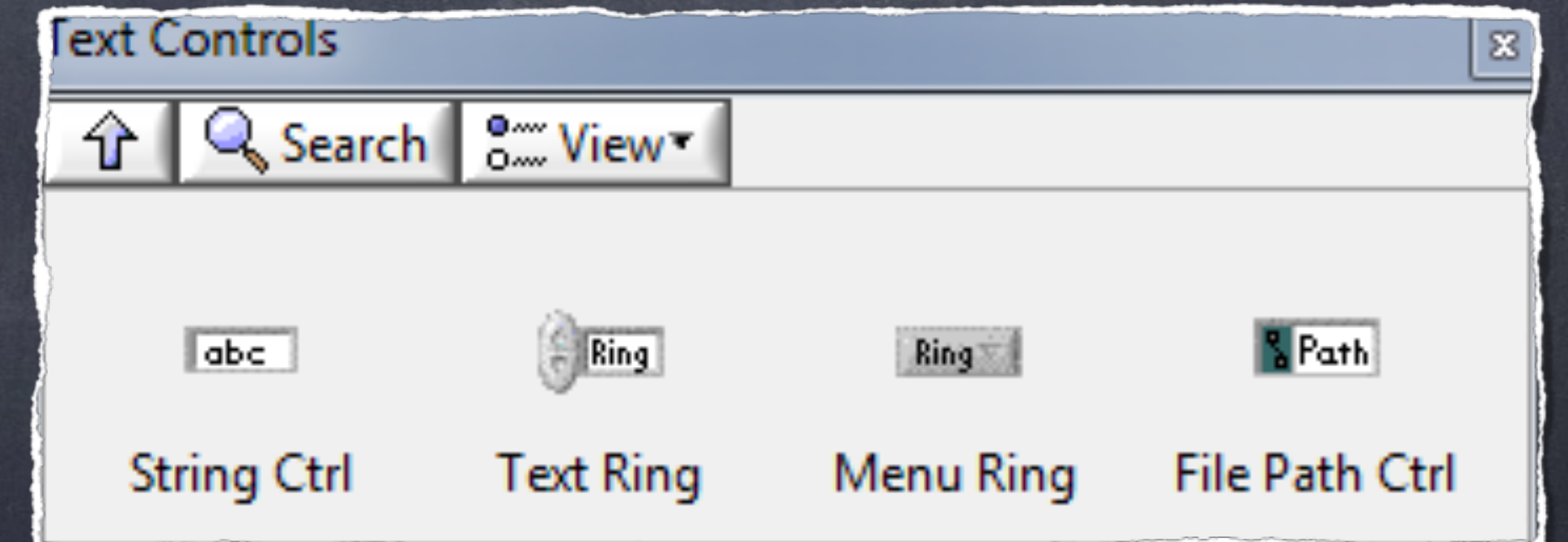
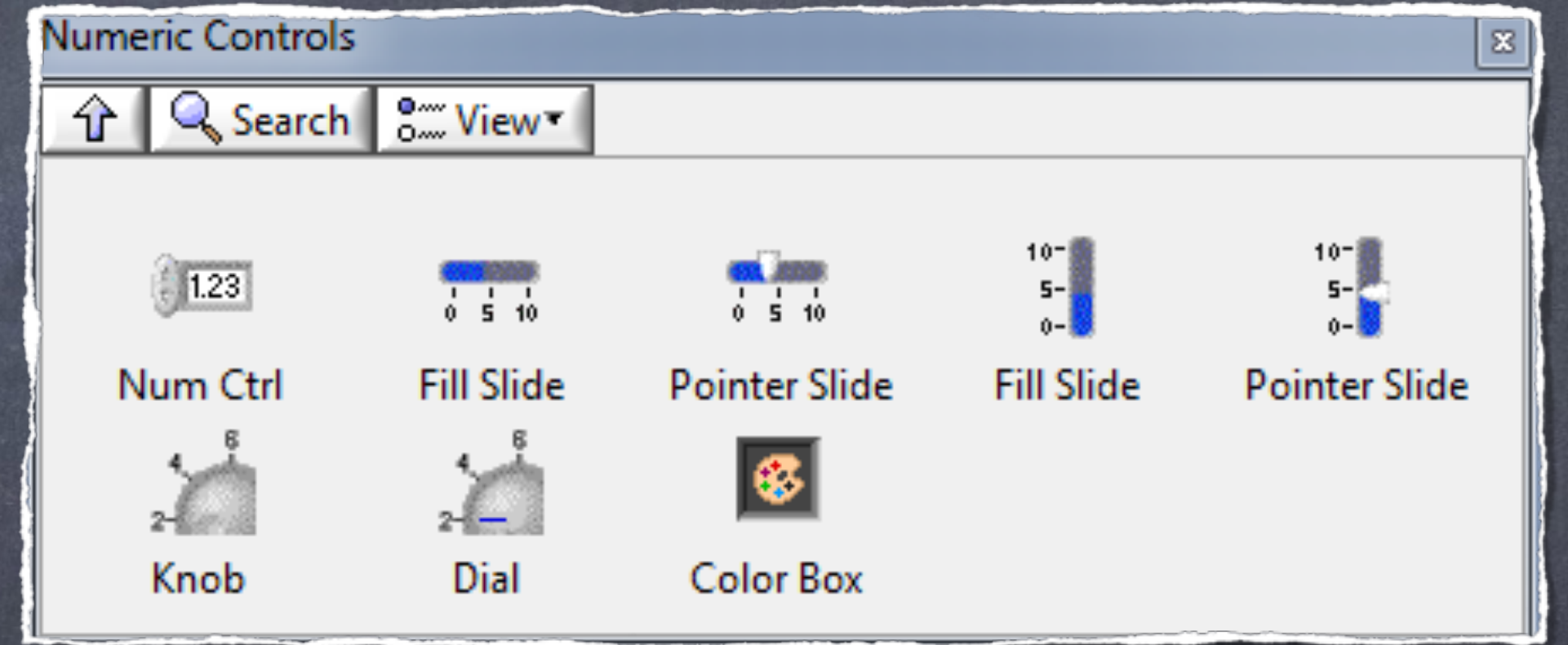
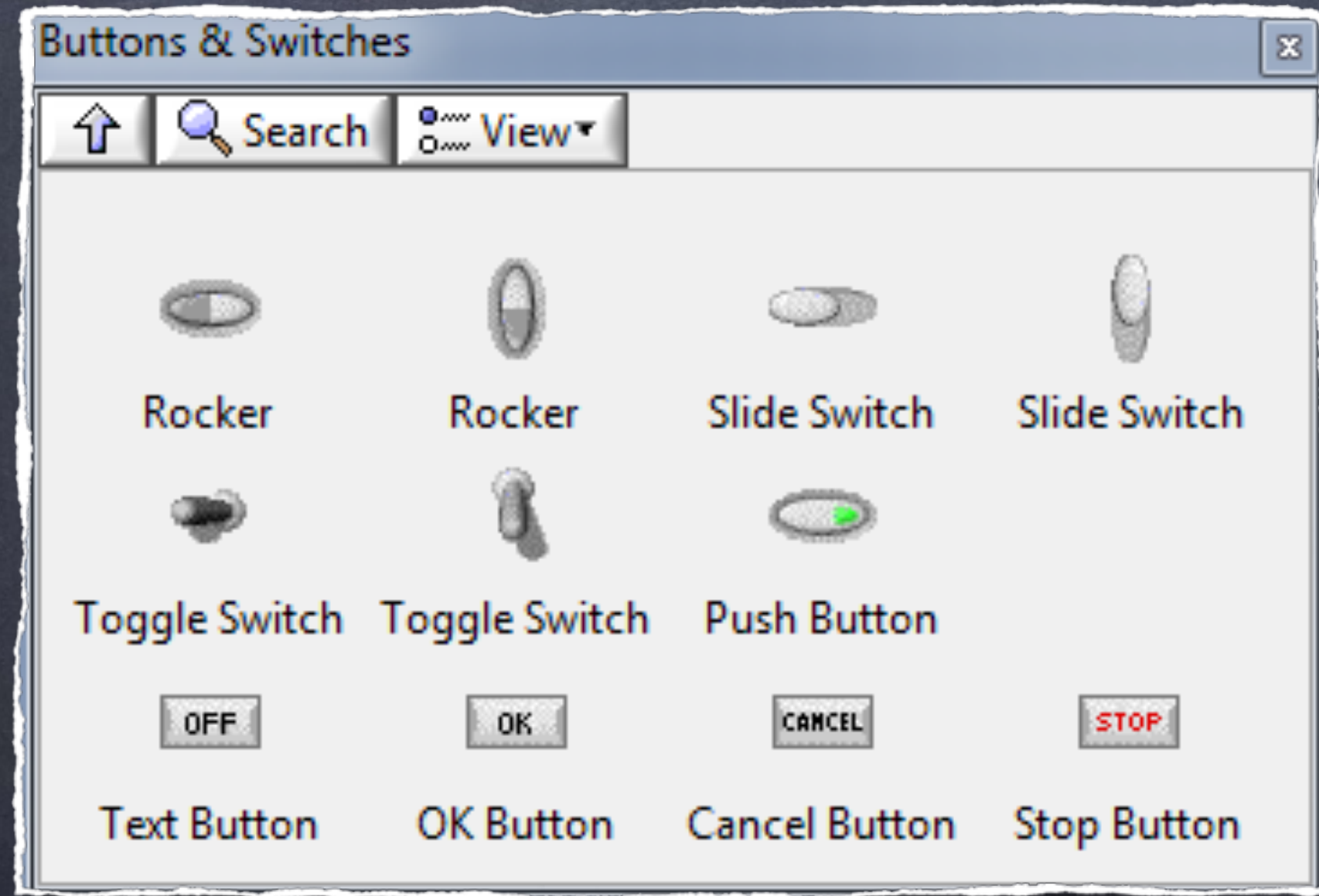




# Tahtalar

## -Yönlendirme Alt Tahtası-

- VI'nın girdilerini ayarlamak için kullanılacak yönlendirmeler:

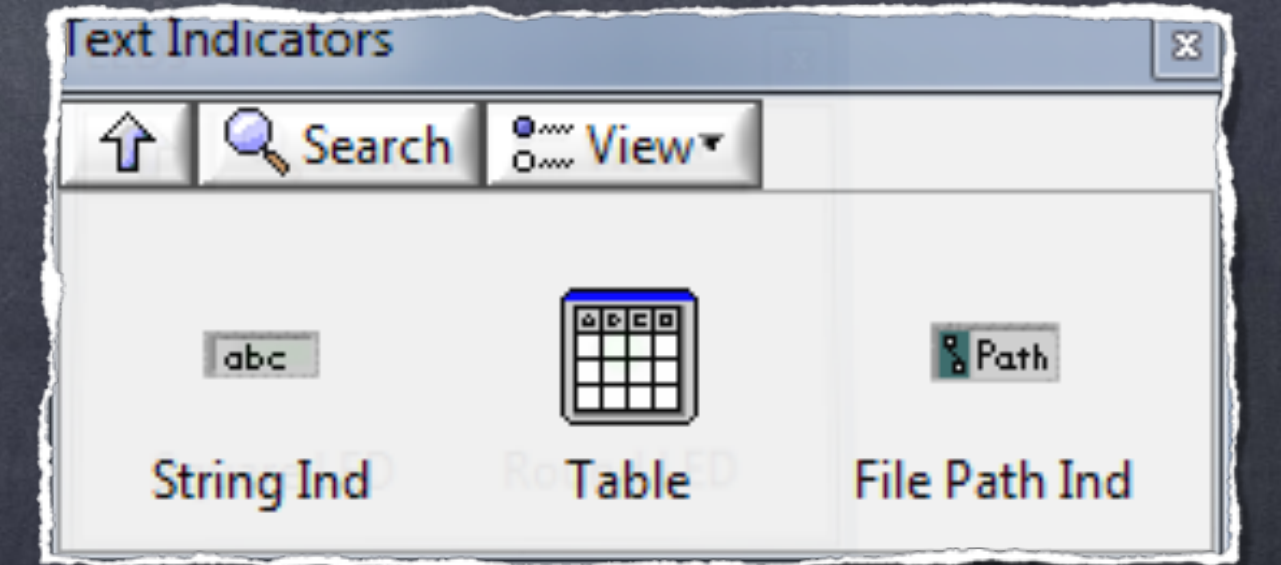
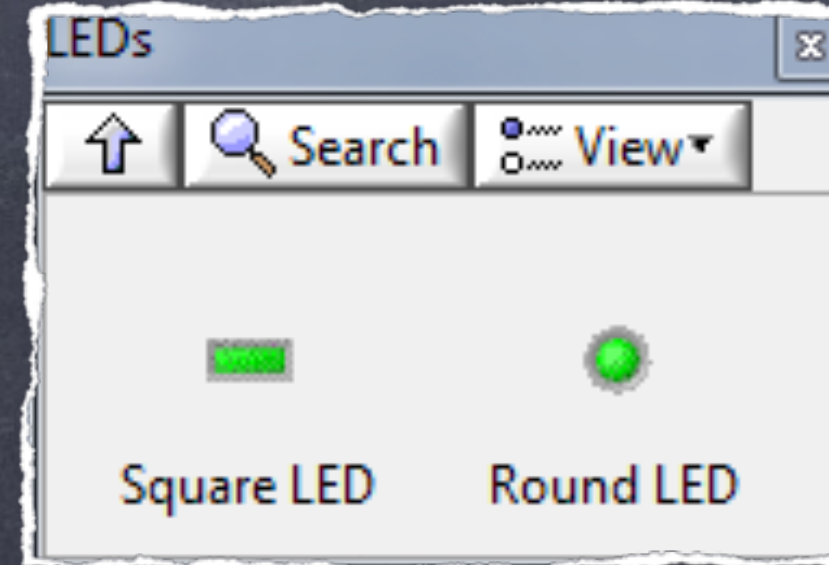
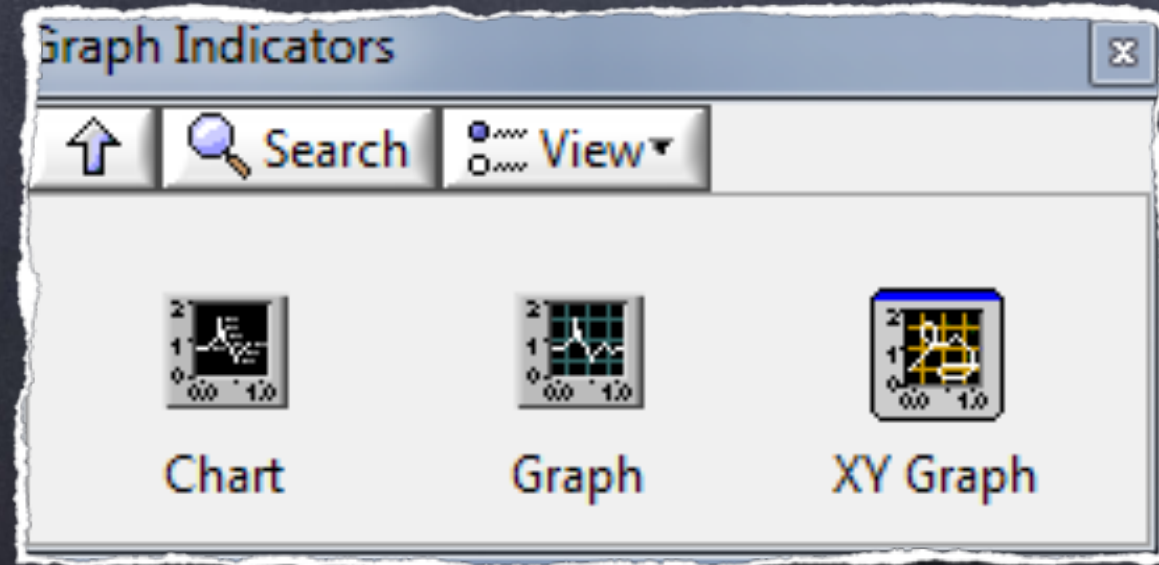
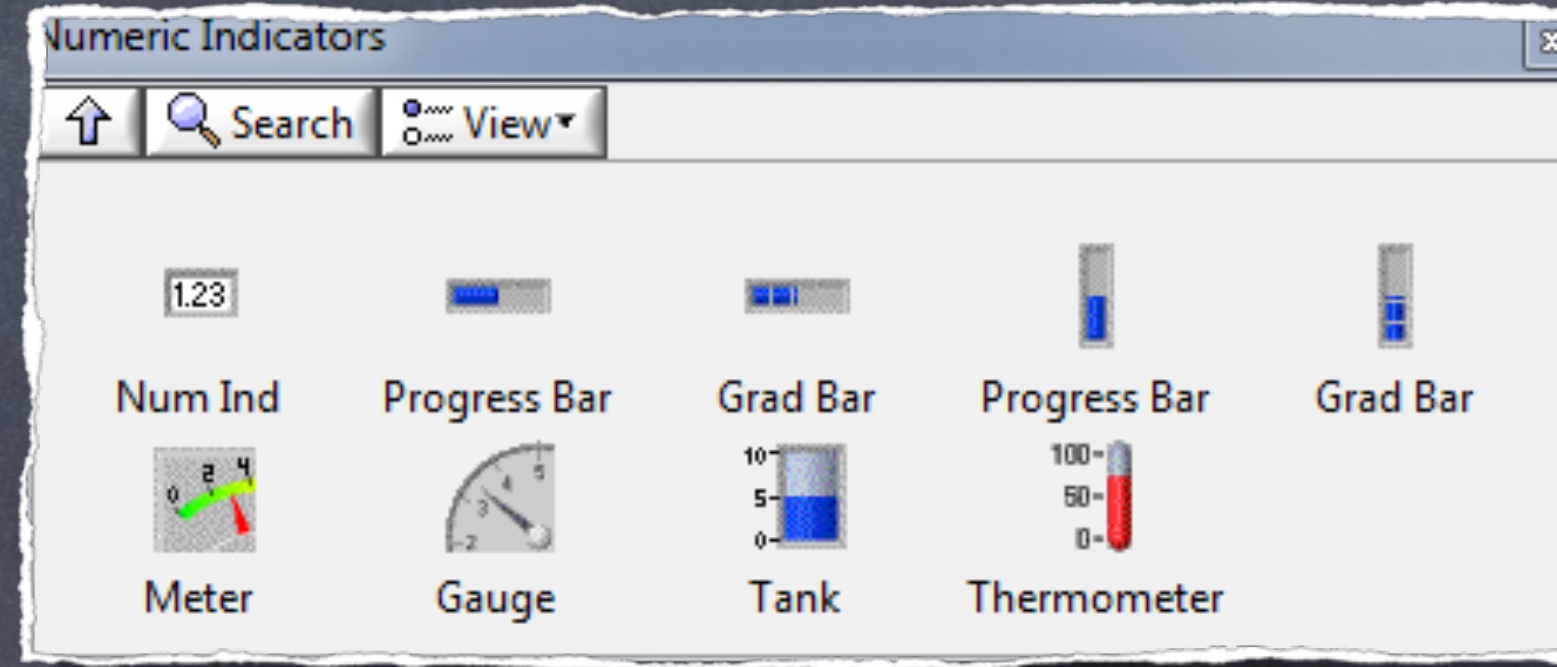




# Tahtalar

## -Göstergeler Alt Tahtası-

•VI'nin çıkışlarını ayarlamak için kullanılacak göstergeler:

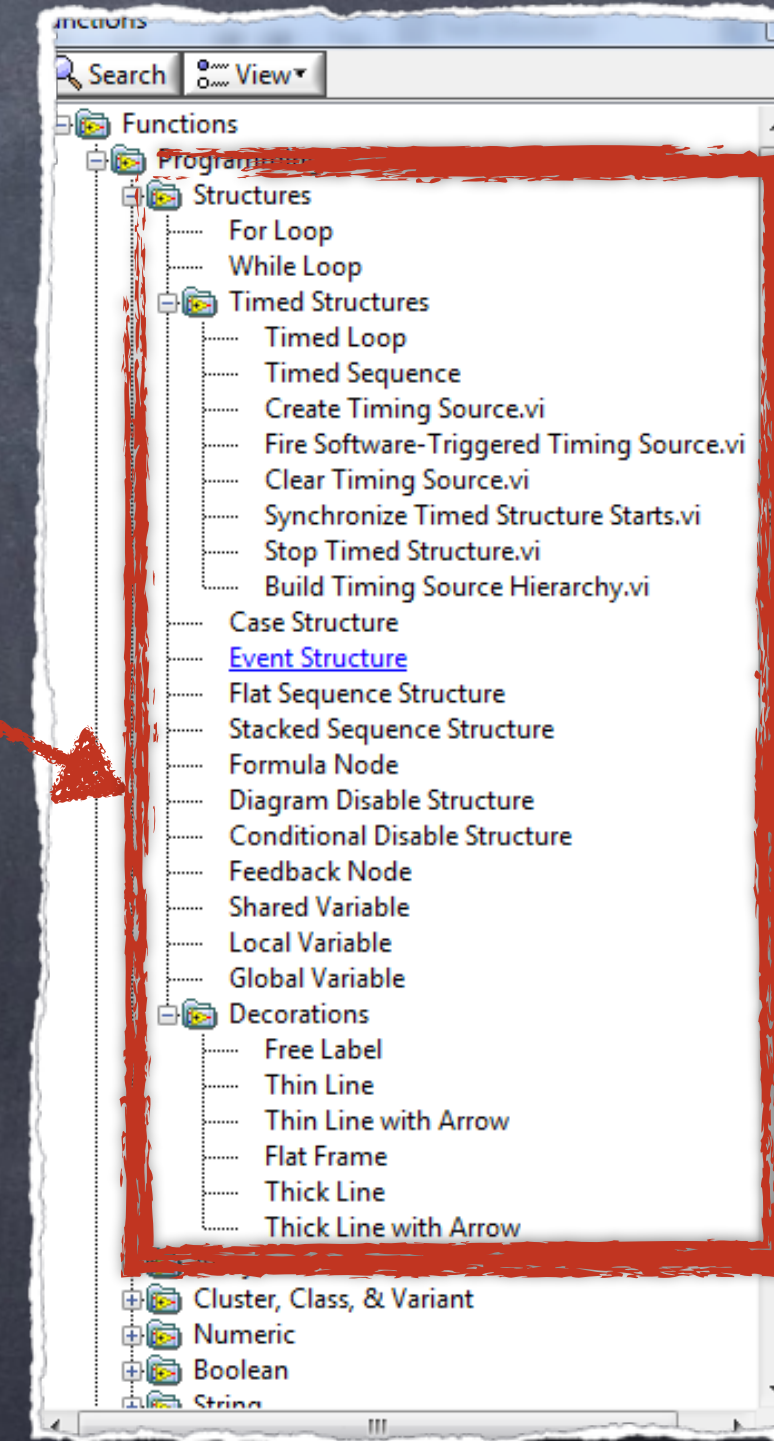
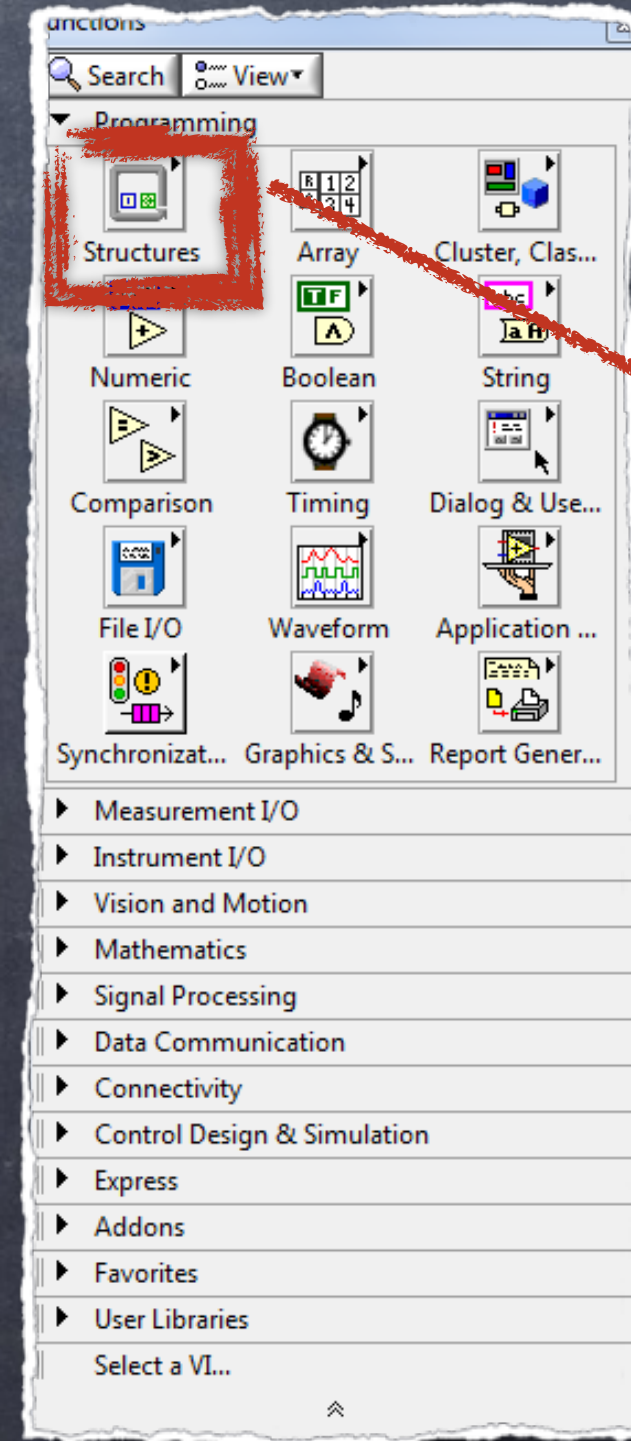




# Tahtalar

## -İşlevler Tahtası-

- VI yapısını oluştururken gerekli olan programlama araçlarını barındırır.
- Akış çizelgesi üzerinde sağ tıklanarak ulaşılabilir.

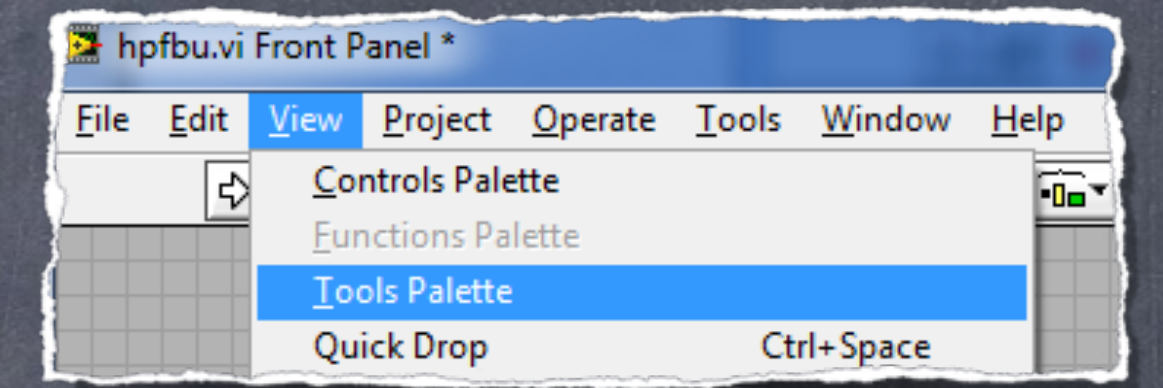
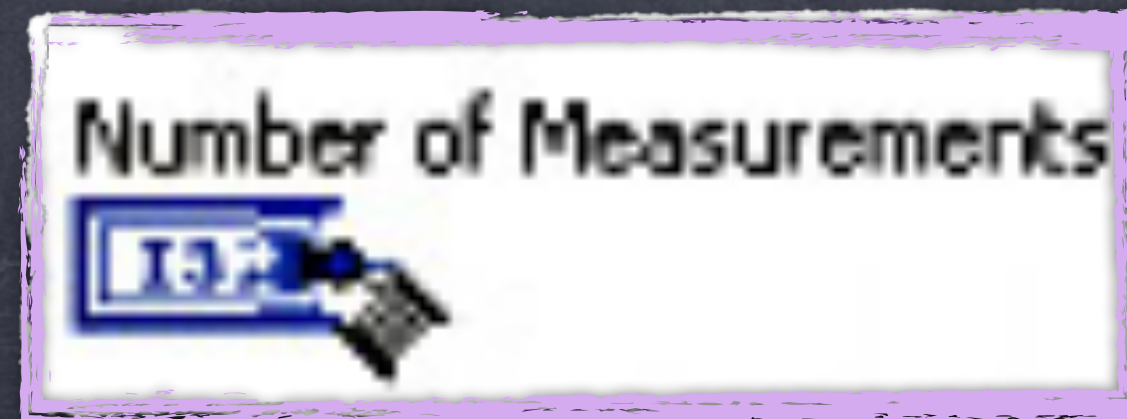
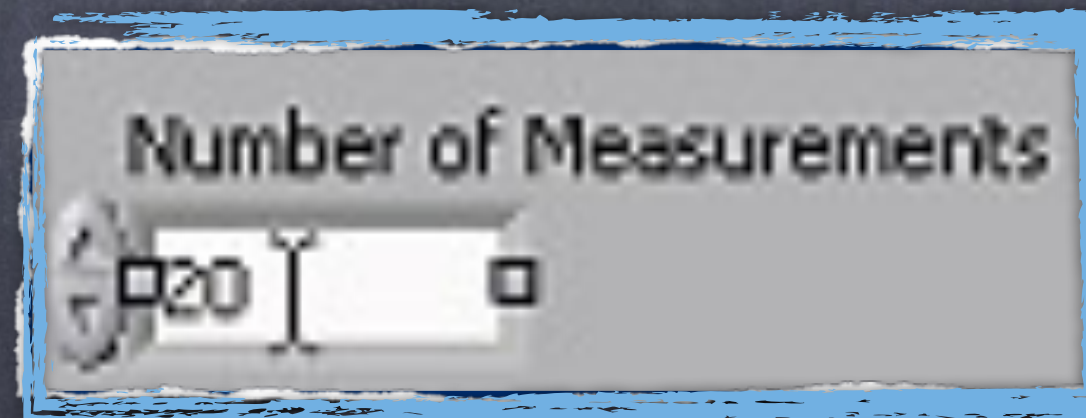
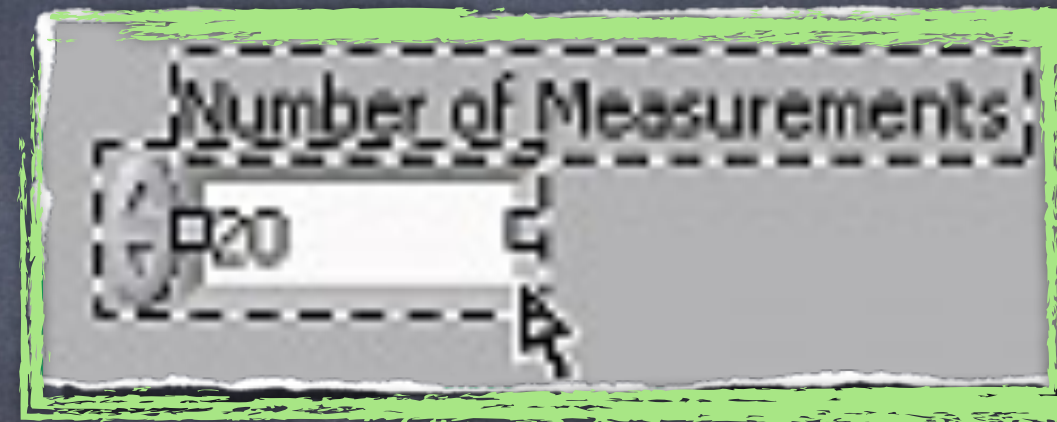
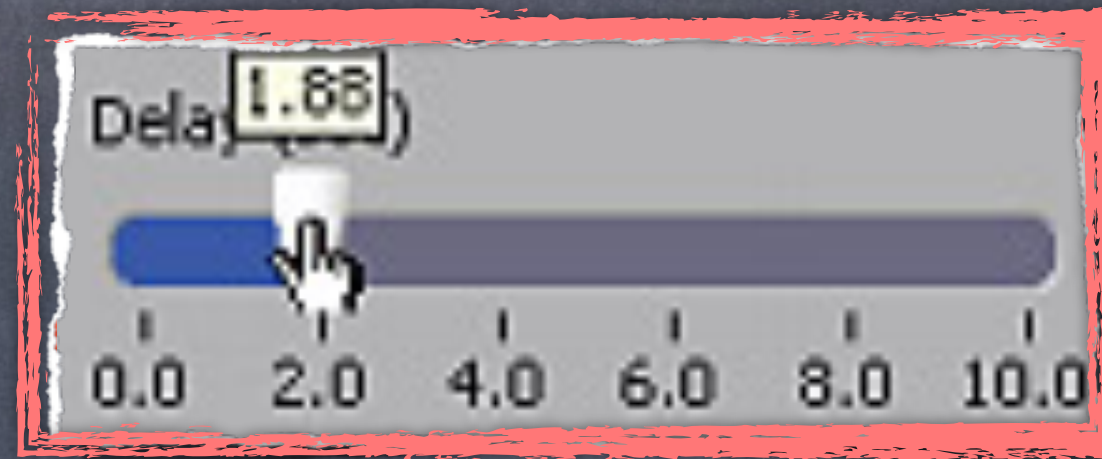




# Tahtalar

## -Araçlar Tahtası-

•VI oluştururken fare imlecinin hangi görev için kullanılacağını ayarlamak için kullanılır.



Kendiliğinden seçim



İşlem aracı

Konumlandırma aracı

Etiketleme aracı

Tel aracı

Kısayol menüsü aracı

Kaydırma aracı

Durma noktası aracı

Sonda aracı






Renk kopyalama aracı

Renklendirme aracı



# Programlama Ortamı

## -Veri türleri-

	Ondalıklı sayı	0.04, 1.008, vb. -64 bit (~15 hane)
	Tam sayı	2, 321, vb. - 32 bit (-2147483648 - 2147483647)
	Dizgi	giriş, işlem, vb.
	Devinimsel	gerekli olan veri türüne kendini dönüştürebilir (çabuk VI).
	Mantıksal	doğru, yanlış



You have connected two terminals of different types.  
The type of the source is double [64-bit real (~15 digit precision)].  
The type of the sink is string.

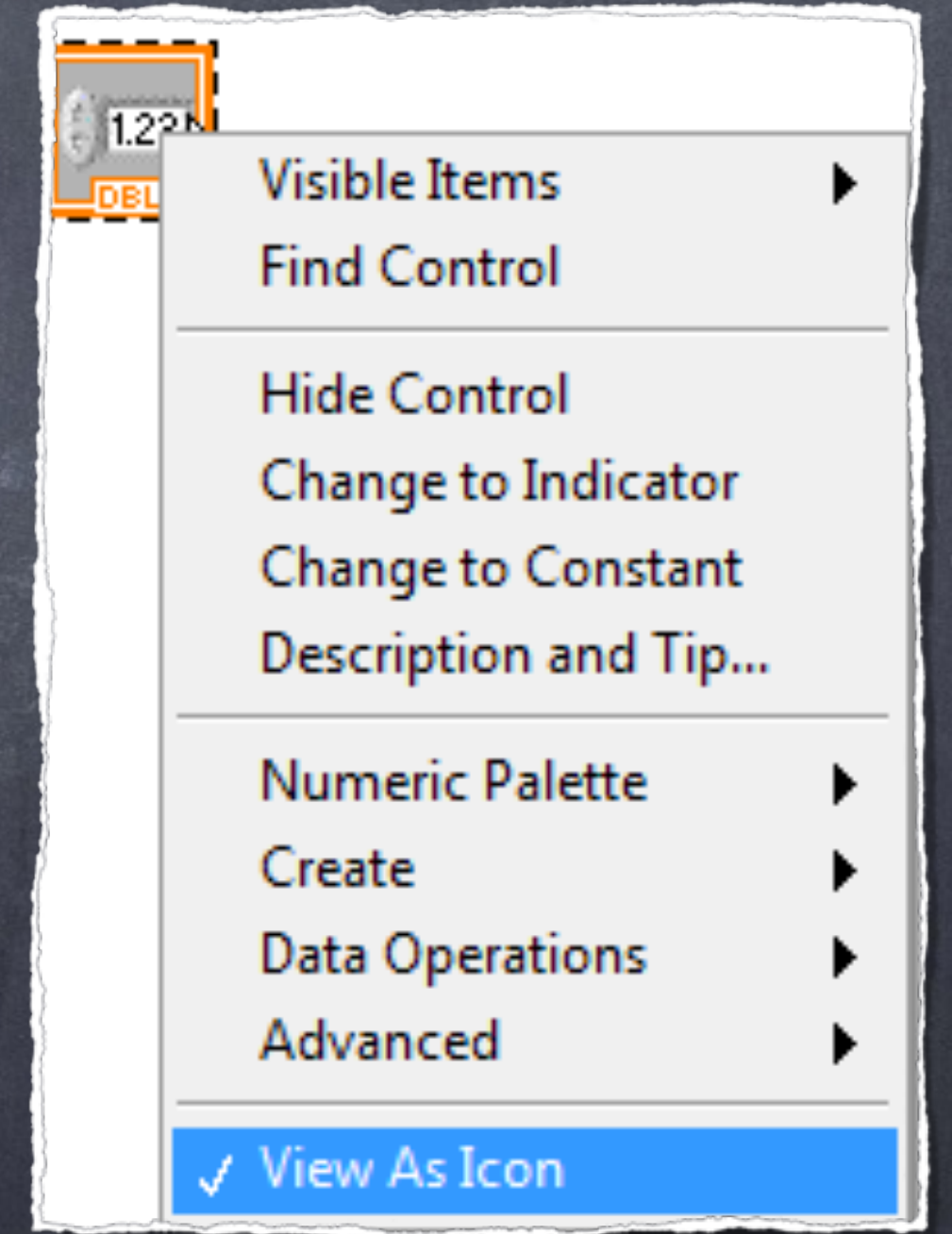


# Programlama Ortamı

## -Uçbirim simgeleri-

- Ön penceredeki bileşenler akış çizelgesinde uçbirim simgeleri olarak gözükürler.
- Girdi ve çıktıları için bağlantı noktalarını ve veri türünü gösterirler.

	Yönlendirme	Gösterge
Uçbirim simgeleri		
Veri türü uçbirim simgeleri		





# Programlama Ortamı

## -Teller-

- Akış çizelgesindeki iki bileşen arasında veri alış-verişi sağlar.
- Tellerin rengi ve kalınlığı veri türleri ile ilgili bilgi verir.

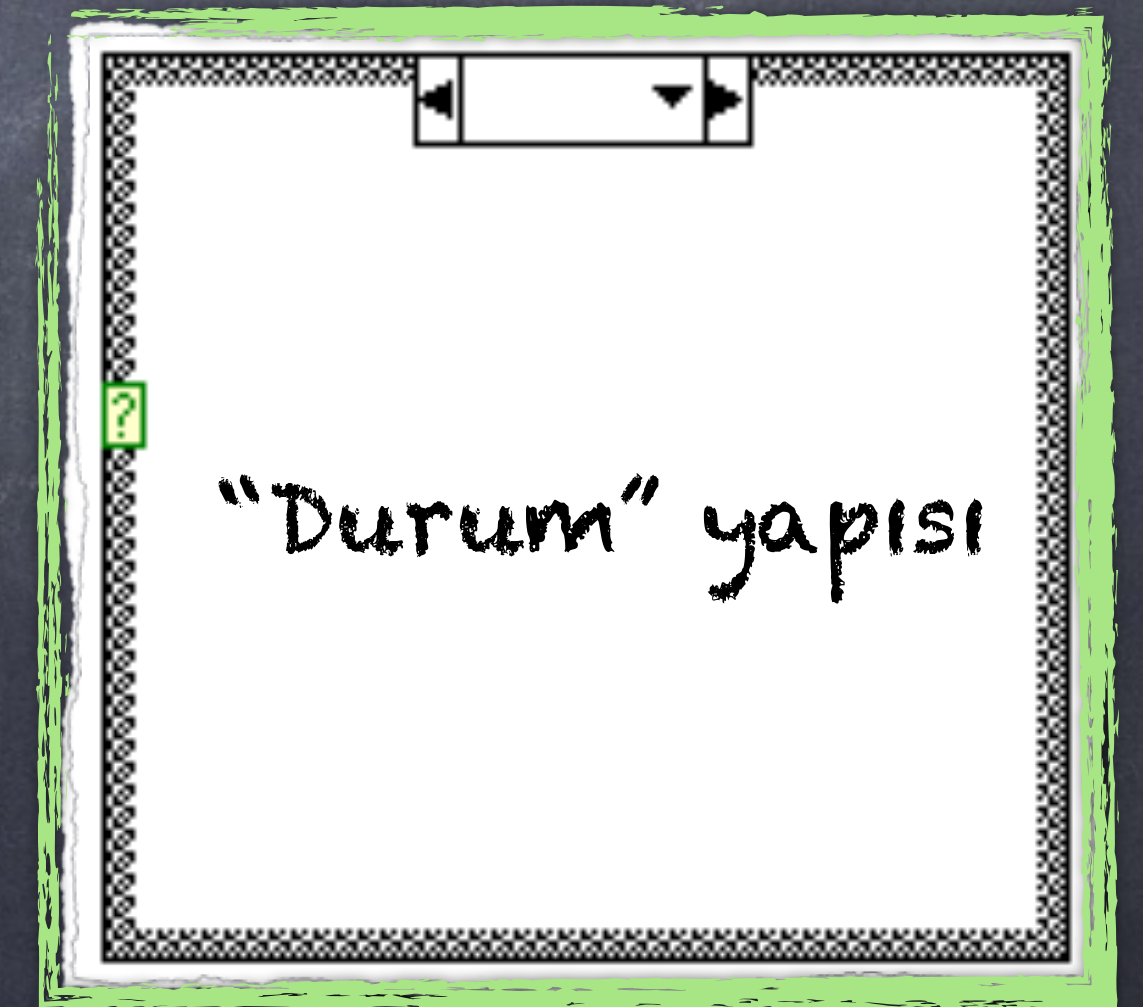
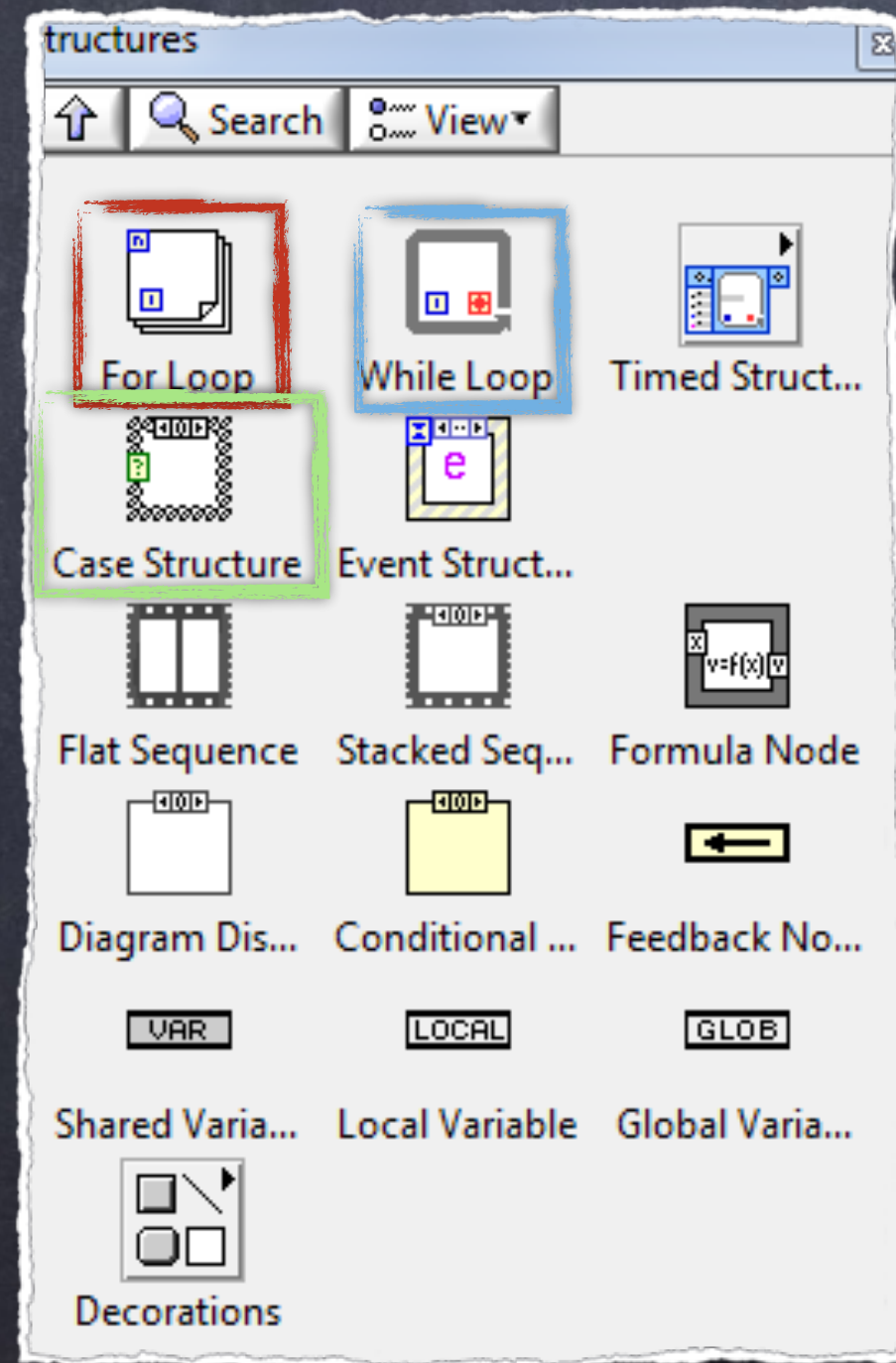
Tel türü	Sayı	1B Dizi	2B Dizi	Renk
Sayısal				Turuncu (ondalıklı) Mavi (tamsayı)
Mantıksal				Yeşil
Dizgi				Pembe



# Programlama Ortamı

## -Yapılar-

- Metin tabanlı programlama dillerindeki yapılara oldukça benzerdir.
- Akış çizelgesinde yapılar görsel olarak boyutu ayarlanabilir çerçeveler ile gösterilirler.
  - Çerçevenin içine aldığı bileşenler yapı türüne göre tekrarlanabilir yada koşula bağlanabilir.



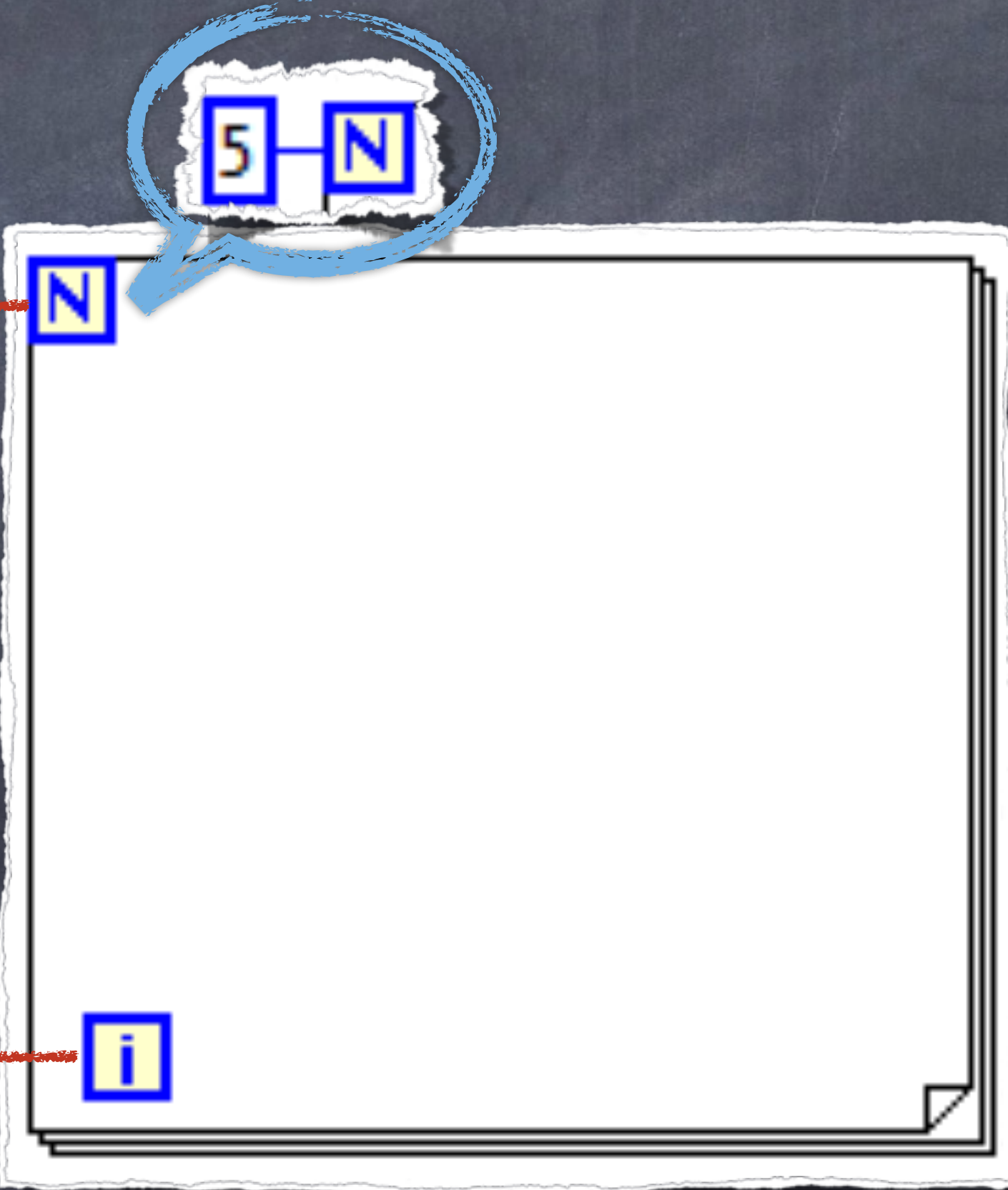


# Yapılar

## -İçin döngüsü-

- Bu döngü VI içindeki bir veya birden fazla işlevin belirtilen sayıda tekrarlanması için kullanılır.

Tekrarlama sayısı ●



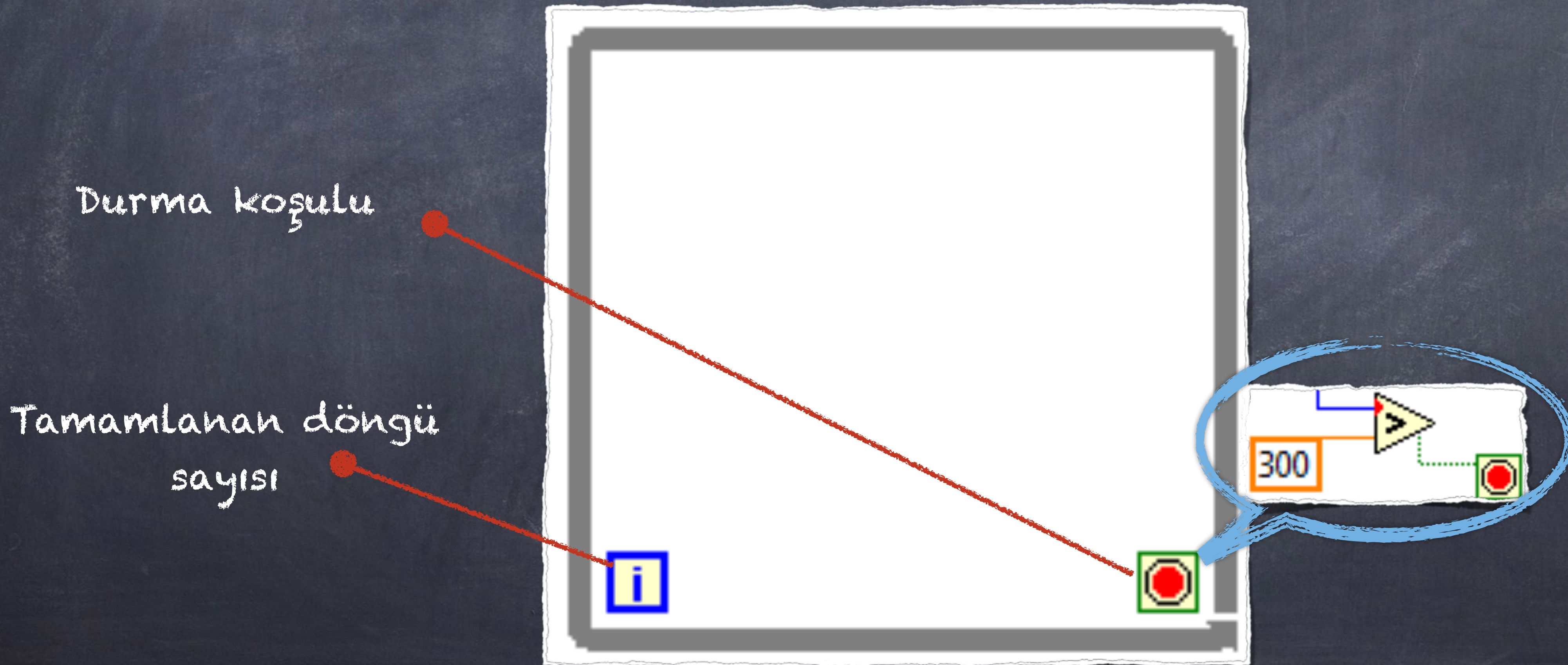
Tamamlanan döngü sayısı ●



# Yapılar

## -İken döngüsü-

- Bu döngü VI içindeki bir veya birden fazla işlevin belirli bir koşul altında tekrarlanması için kullanılır.





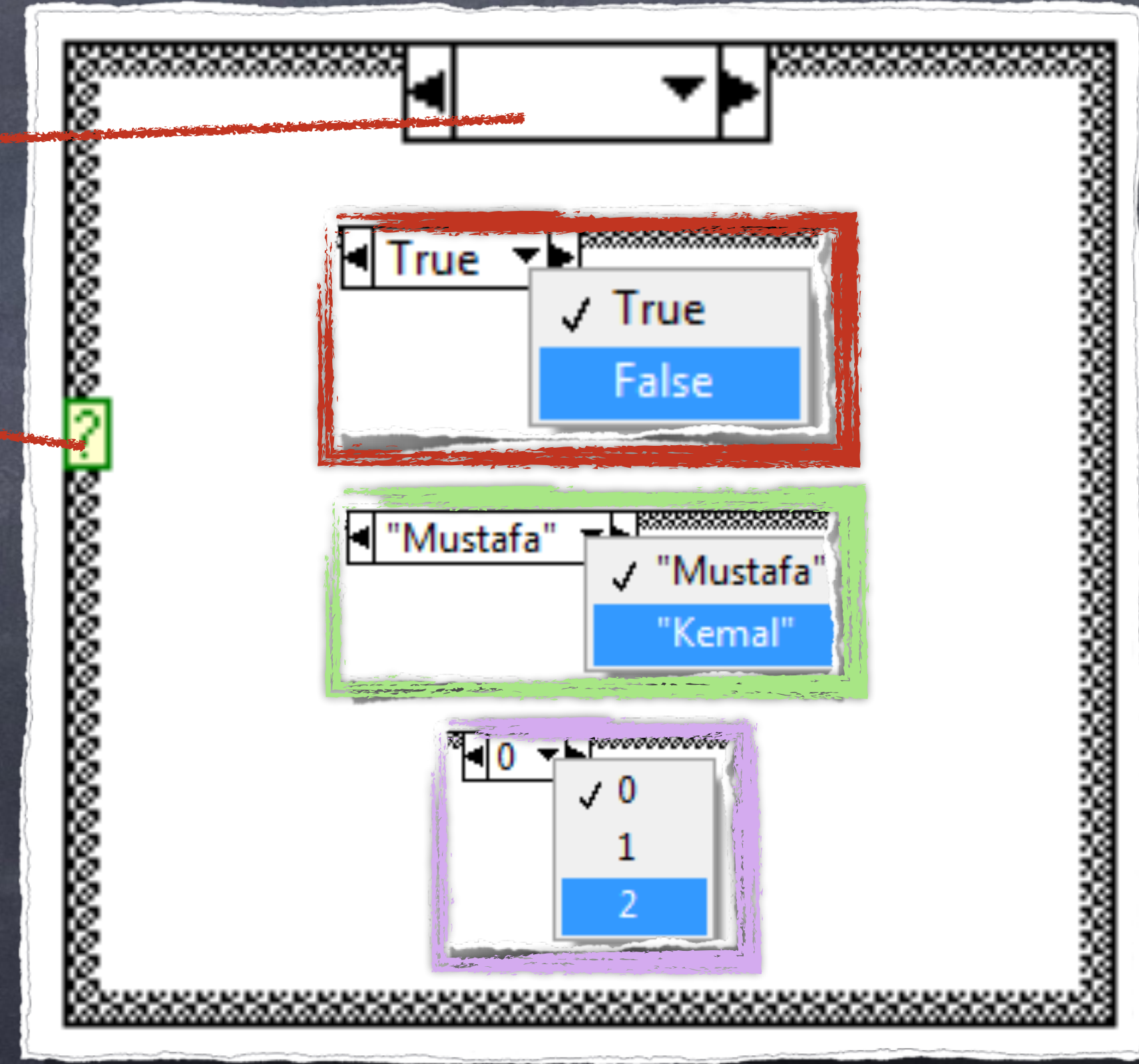
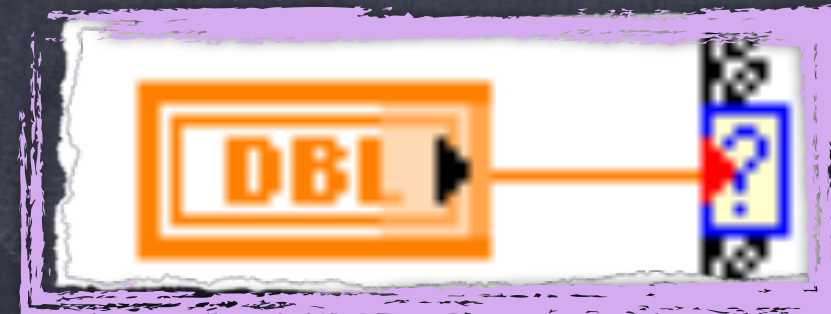
# Yapılar

## -Durum yapısı-

- Bu yapı VI içindeki bir veya birden fazla işlevin farklı durumlar altında nasıl çalışacağını belirlemek için kullanılır.

Durum seçici

Seçim uçbirimi

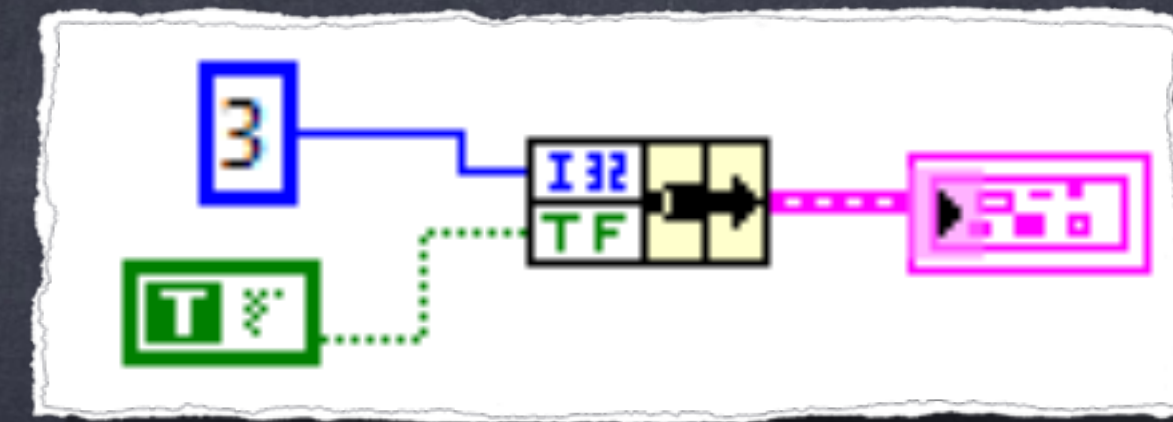
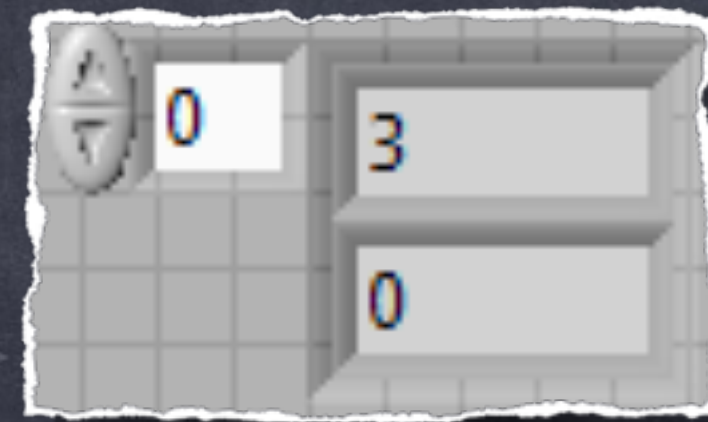
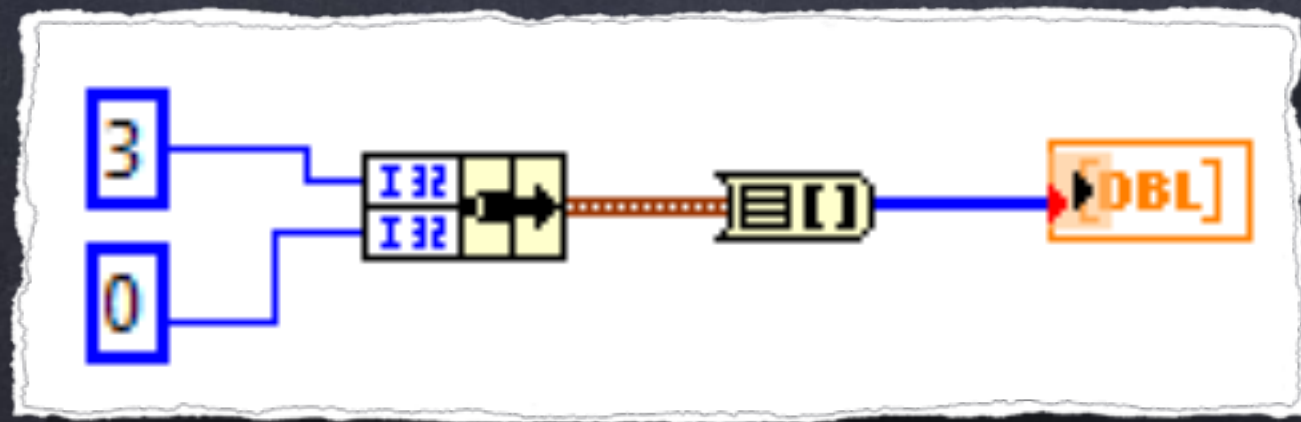




# Programlama Ortamı

## -Dizi ve kümeler-

- Metin tabanlı programlama dillerinde olduğu gibi diziler aynı veri türündeki bir öbek verinin toplandığı bileşenlerdir.
  - Ör: Bir rastgele sayı üreticiden gelen verilerin çıktısını aynı veri türünden oluştukları için bir dizi olarak depolamak mantıklıdır.
- Kümeler ise (Metin tabanlı programlama dillerindeki sınıflar gibi) farklı veri türündeki verilerin toplanabildiği bileşenlerdir.
  - Akış çizelgesindeki tel sayısını azaltmasının yanı sıra bir çok farklı veri türündeki verinin birleştirilip tek uçbirime girdi olarak verilmesini sağlar.

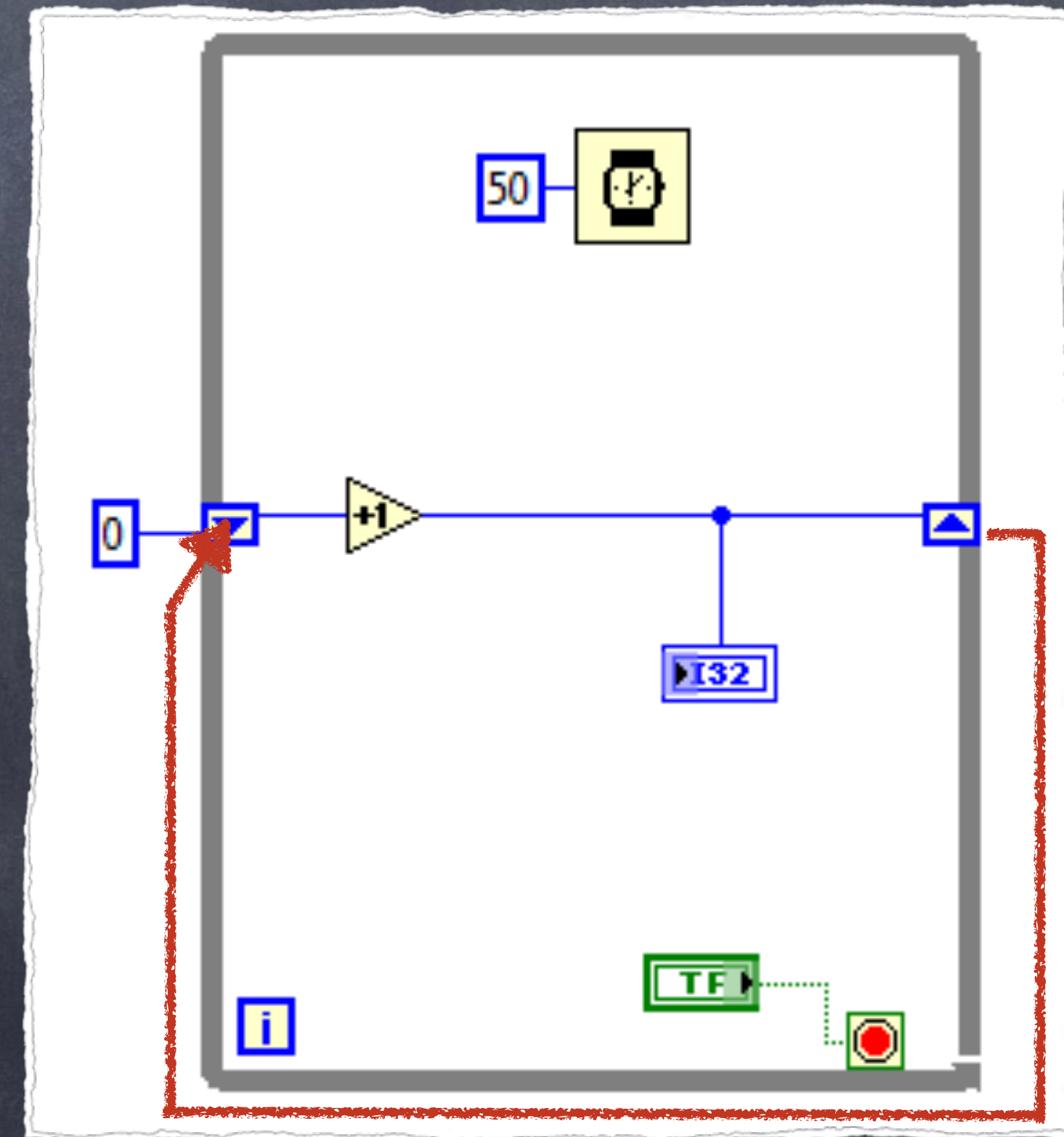
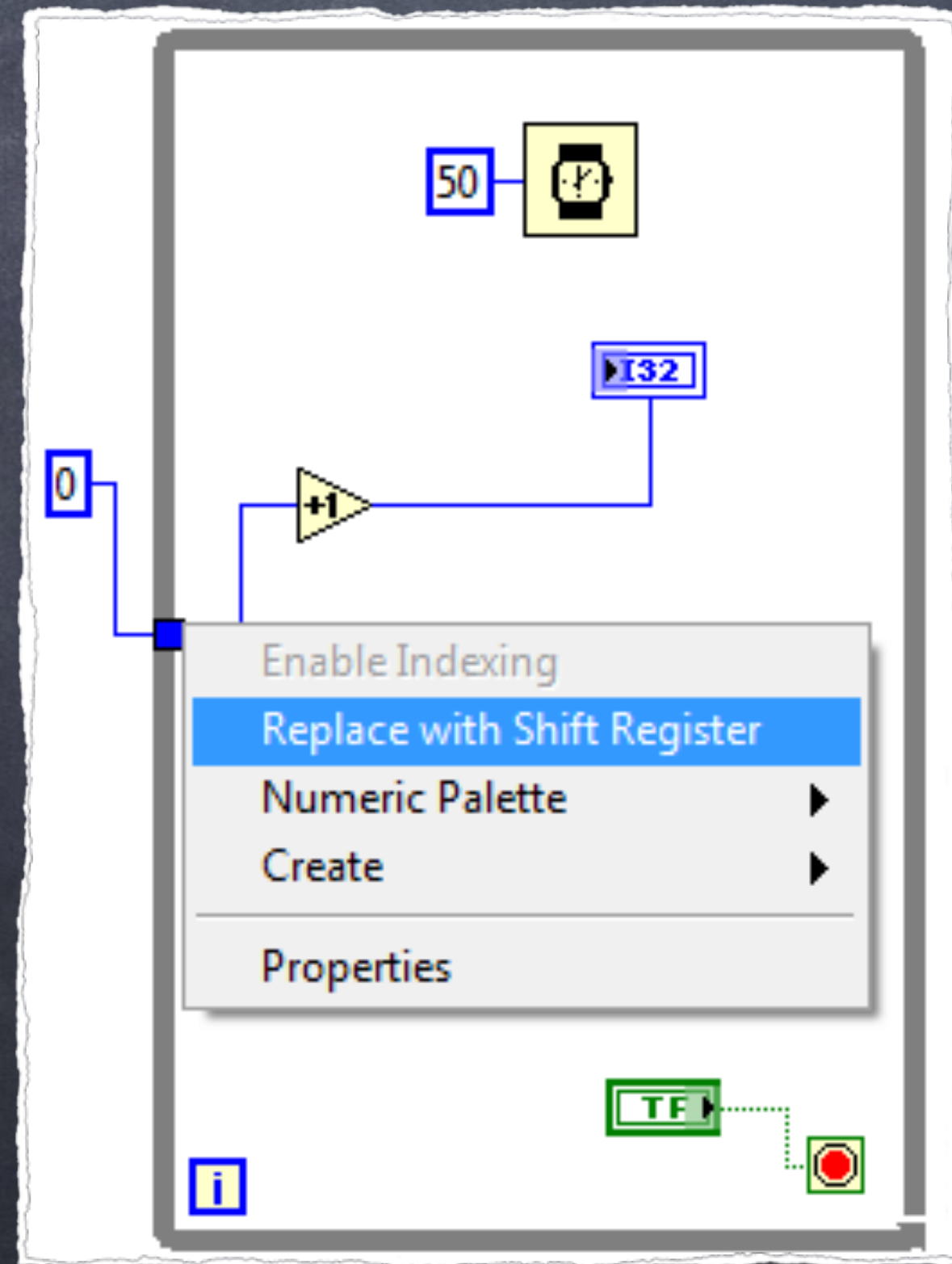




# Programlama Ortamı

## -Kaydırma yazması-

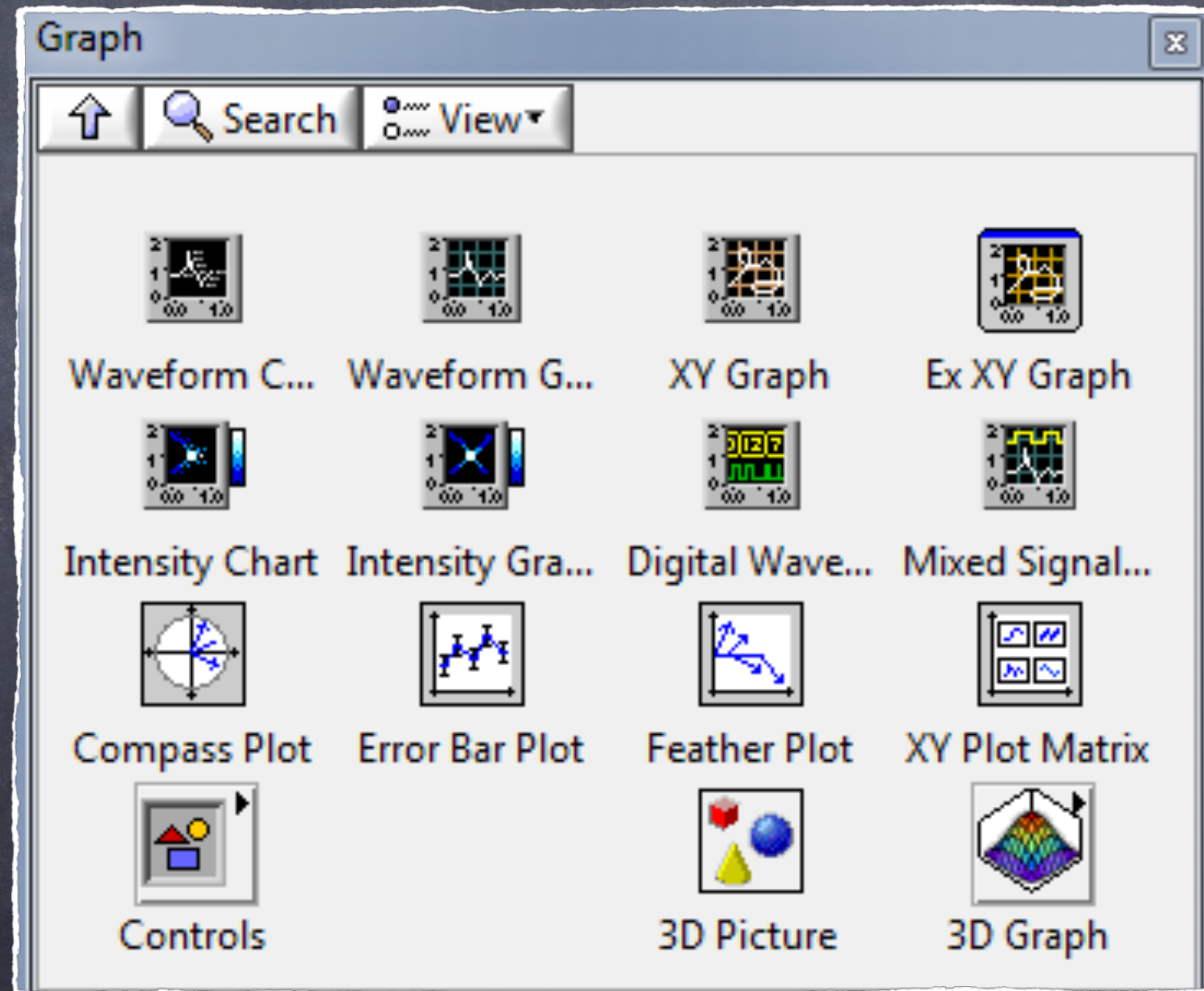
- Döngüler ile çalışırken bir önceki çevrimdeki verileri kaydederek yeni çevrim için kullanılabilir duruma getirir.





# Programlama Ortamı

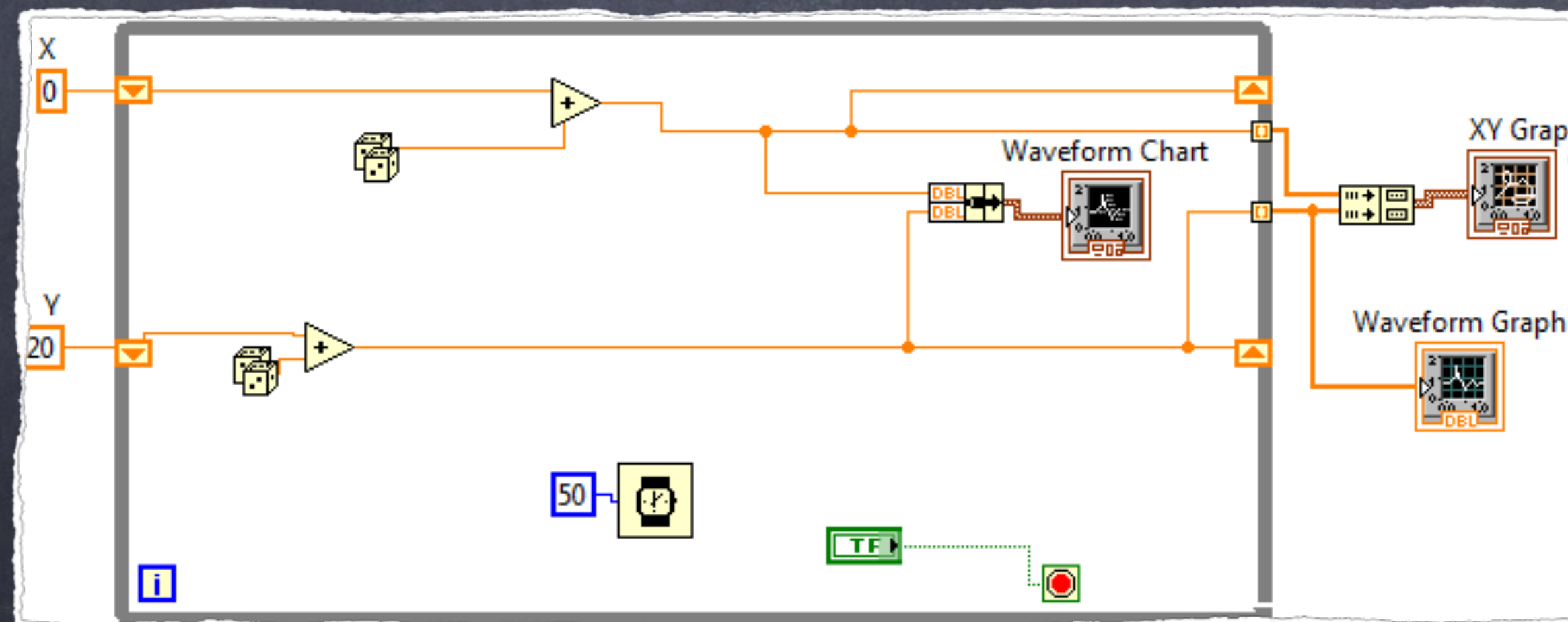
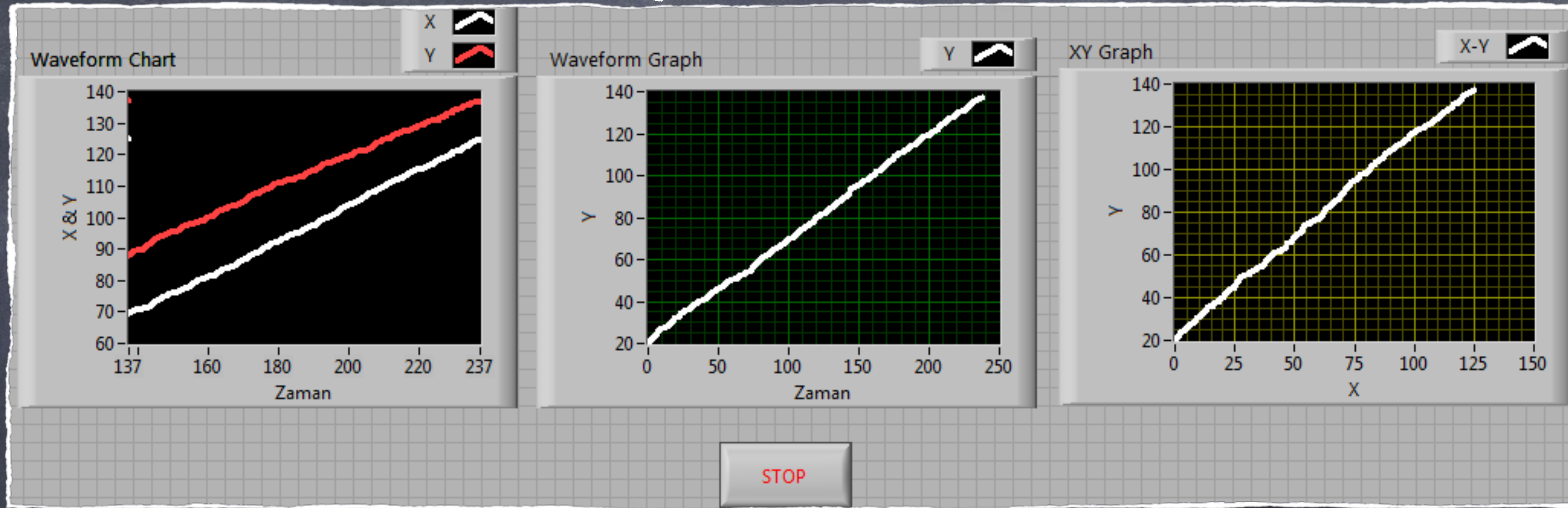
## -Çizimler-





# Programlama Ortamı

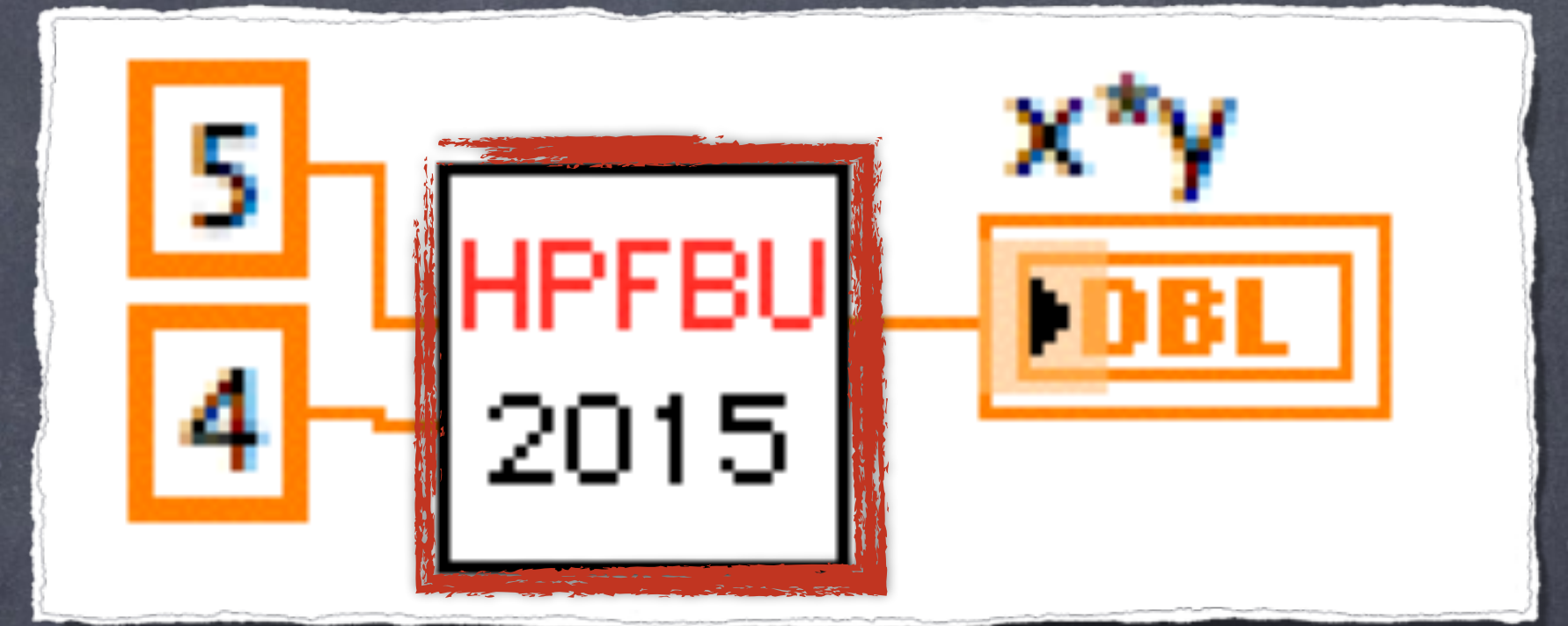
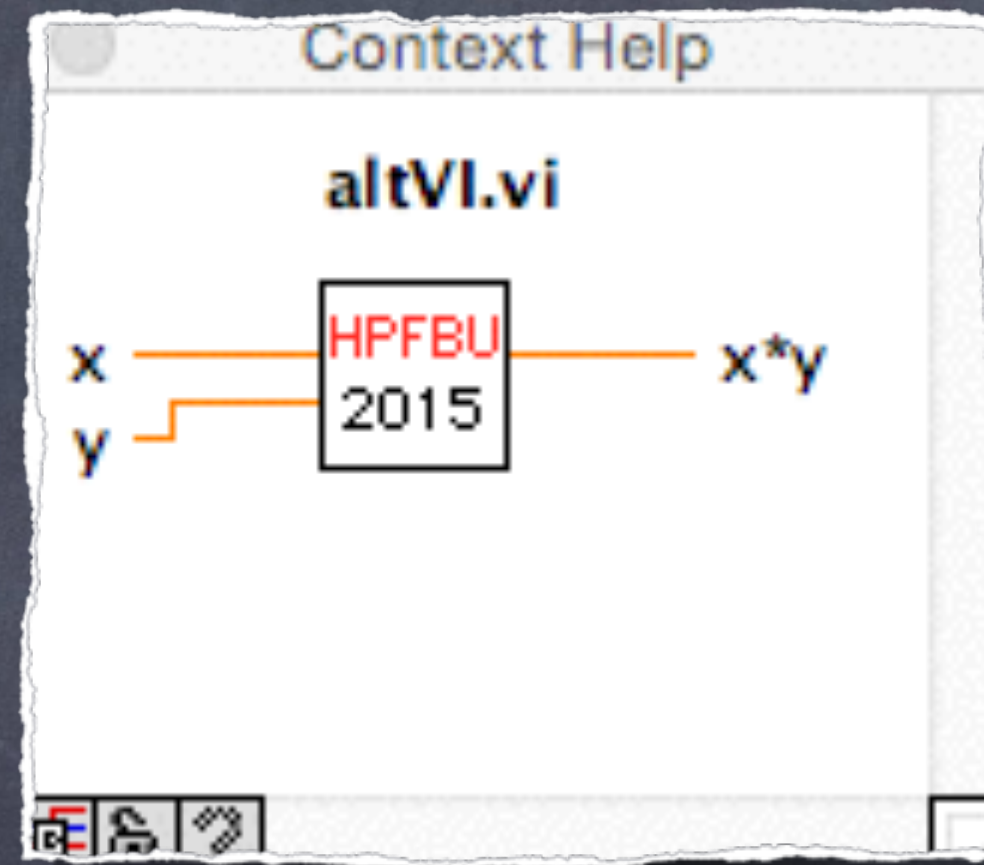
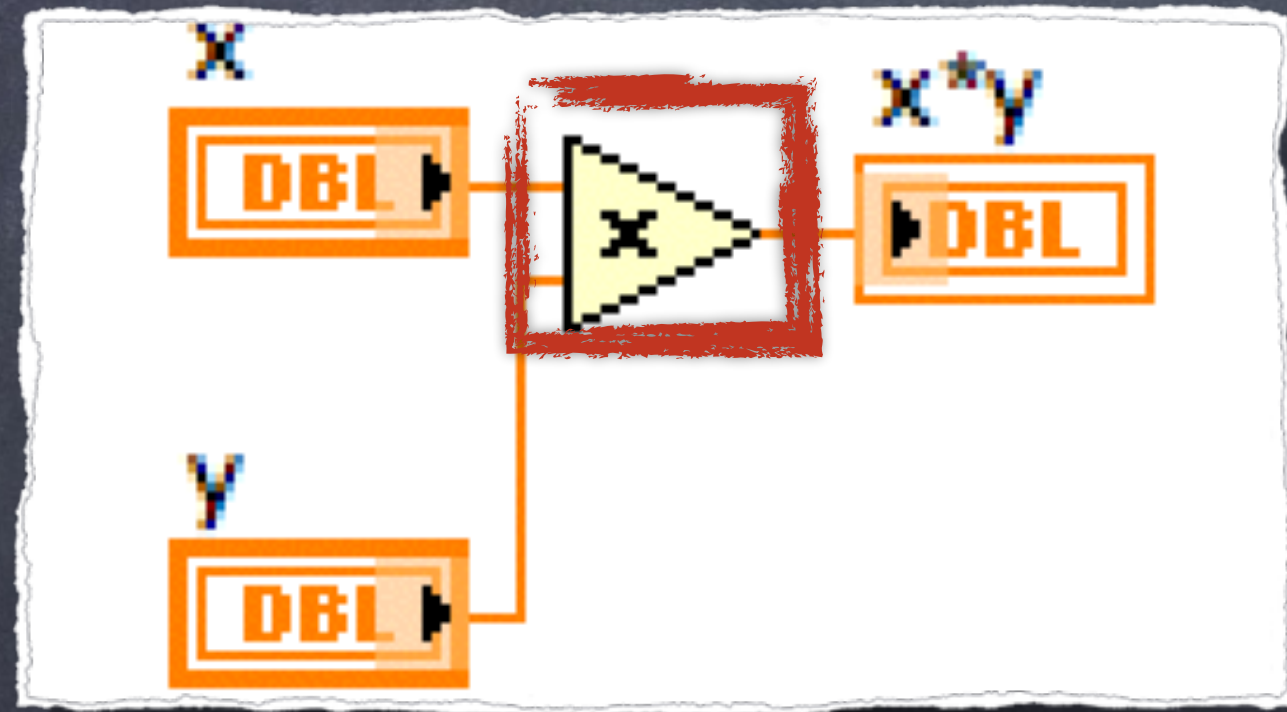
## -Çizimler-





# Programlama Ortamı

-Alt VI-

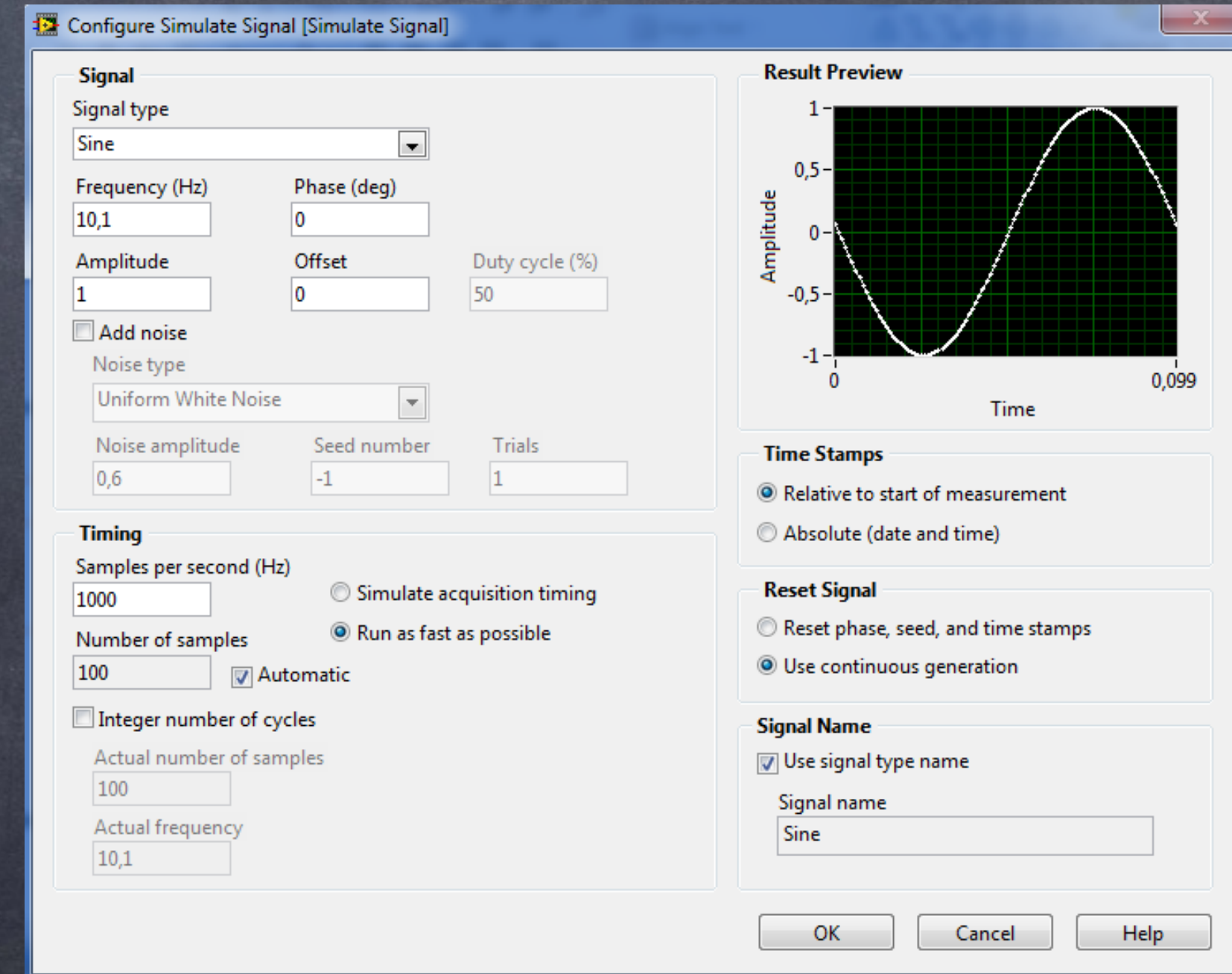




# Programlama Ortamı

## -Çabuk VI-

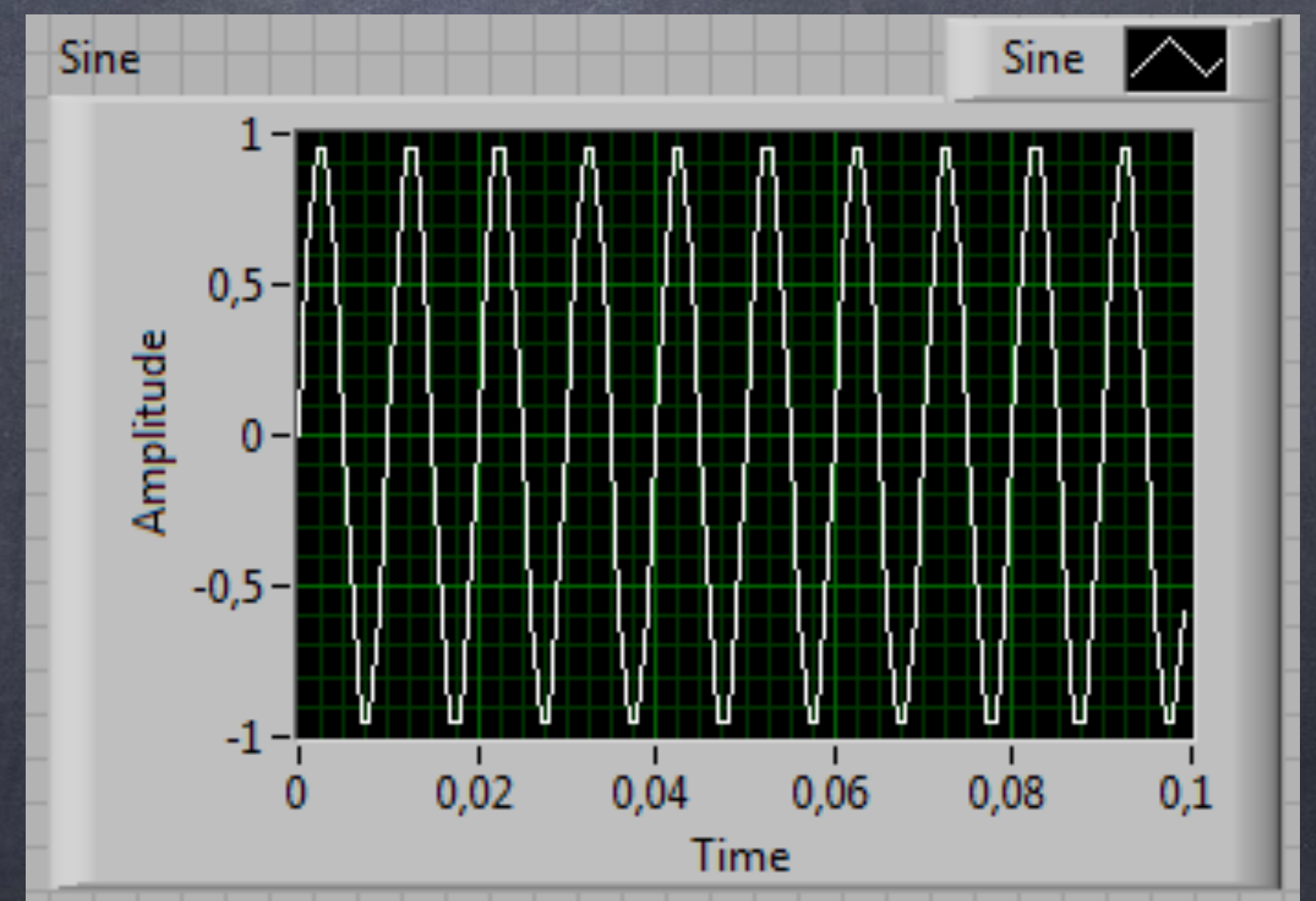
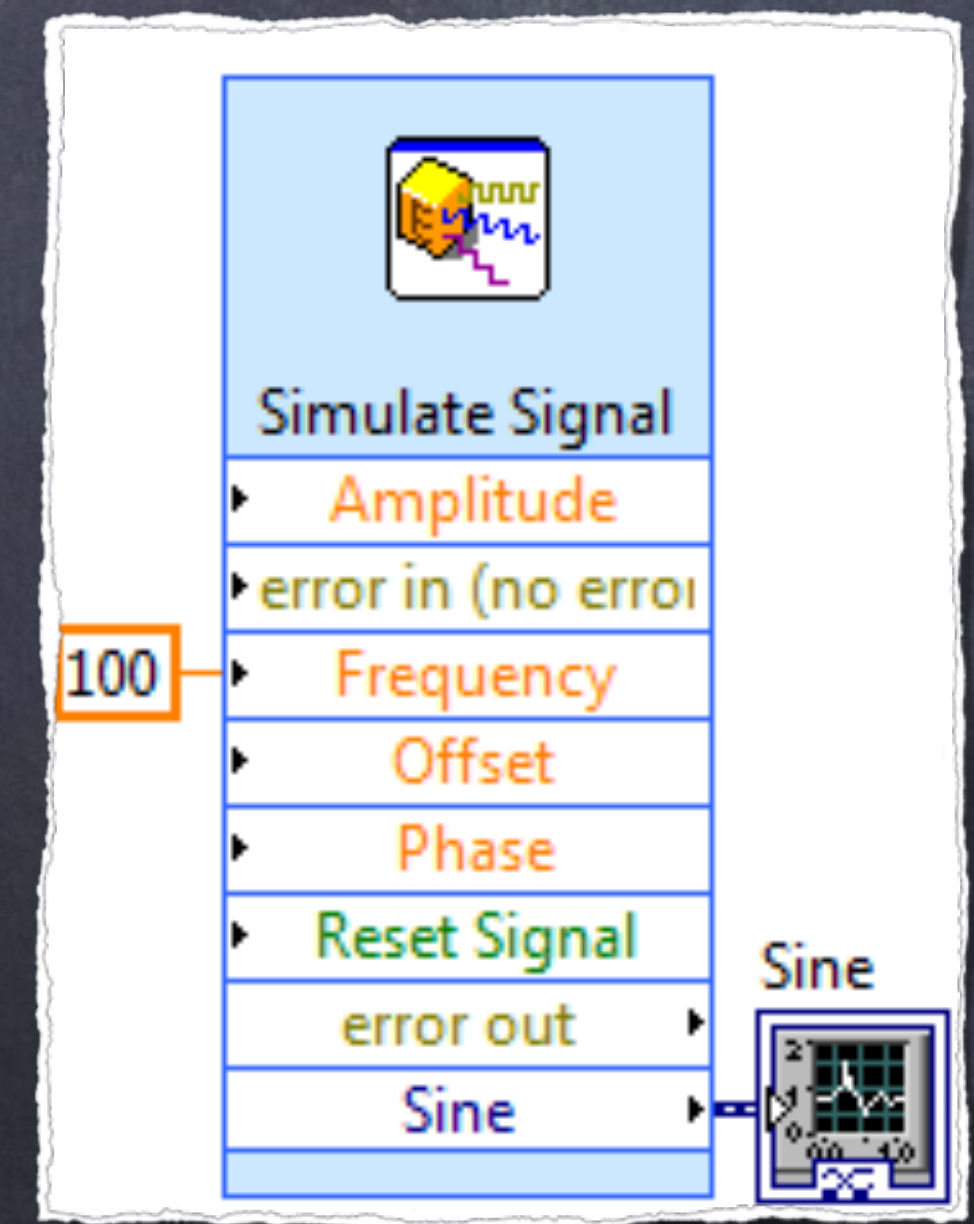
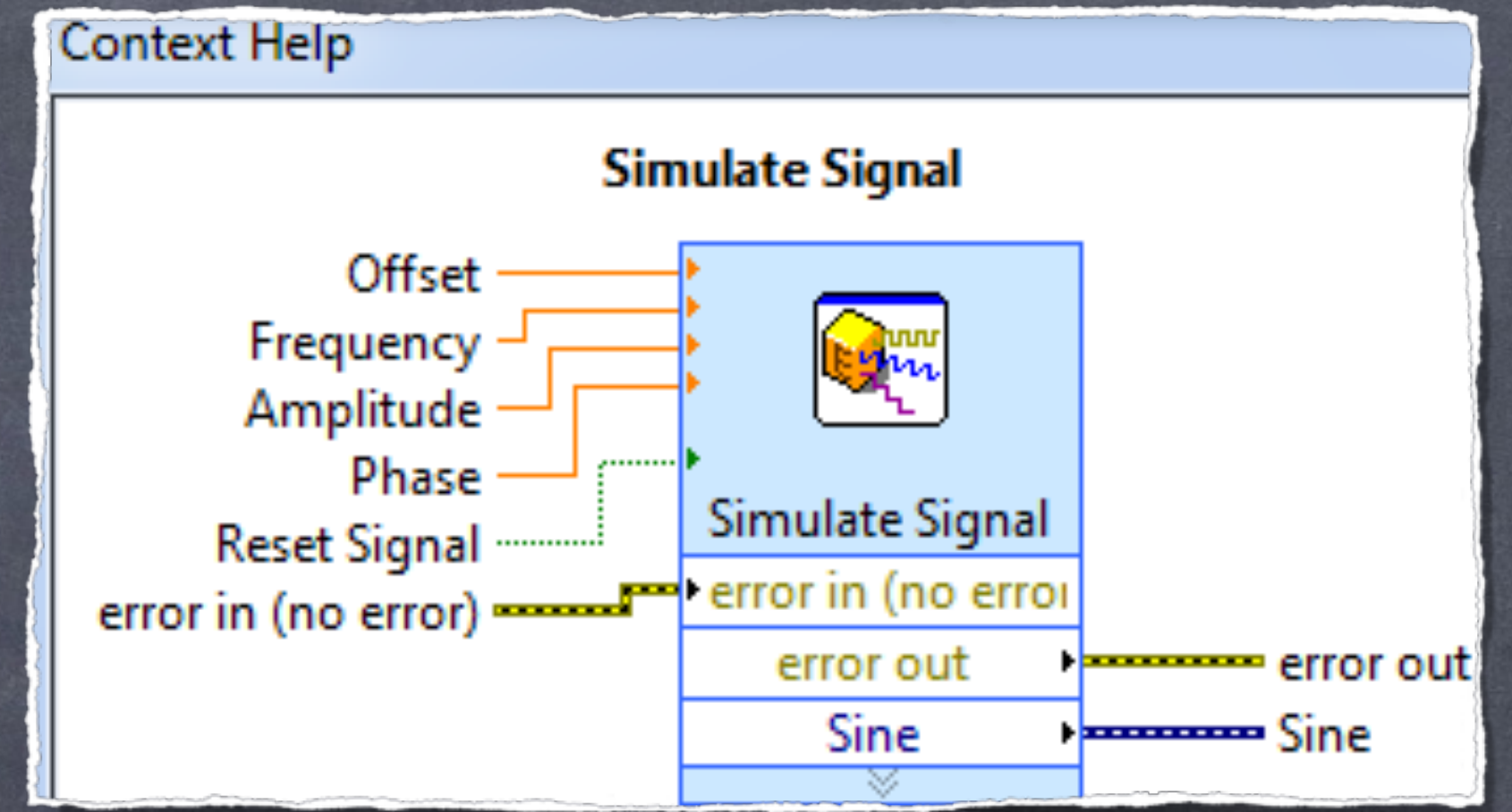
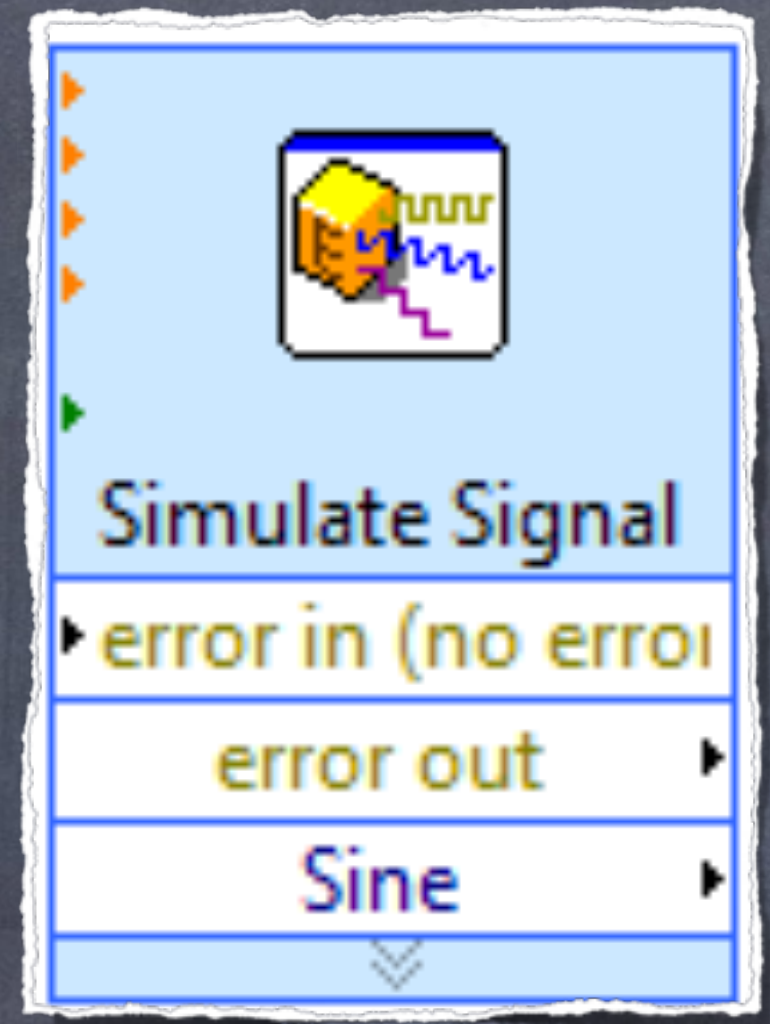
- Program ile hazır gelen ve kullanıcıya hızlı gözümler sağlayan özel VI'lardır.
- Akış şizelgesine eklendiğinde bir yapılandırma penceresi açılarak VI ile ilgili değışkenlerin ayarlanmasına imkan sağlar.
- Böylece aynı işlevi gerçekleştirebilmek için kullanmanız gereken bileşen sayısı ve dolayısıyla tel sayısı azalır.





# Programlama Ortamı

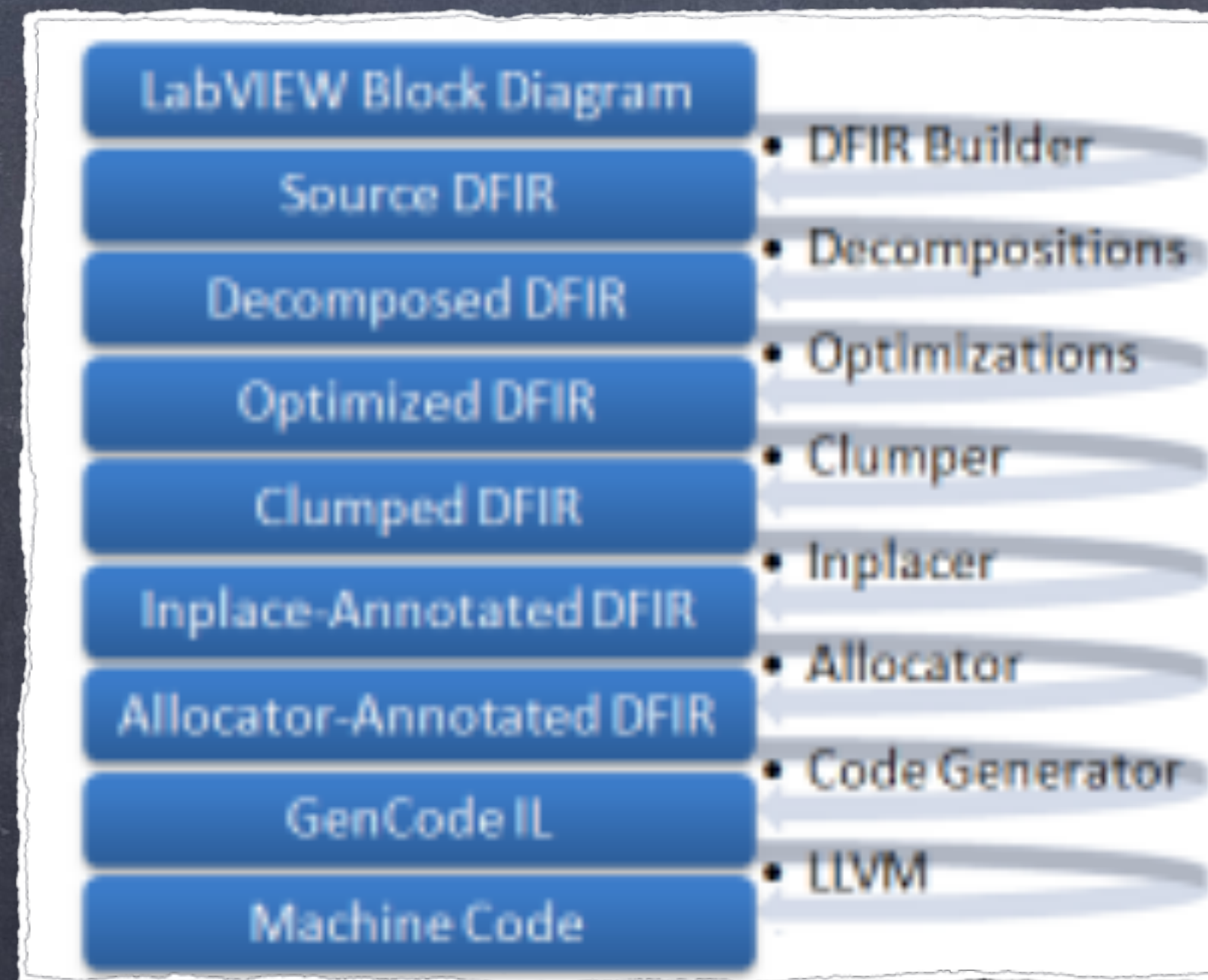
-Çabuk VI-





# VI Derleme

- Oluşturulan programın derlenmesi, VI'yi geliştirmek için "➔" simgesine tıklandığında kendiğiliğinden gerçekleştirilir.
- G derleyicisi akış çizelgesini, geliştirilen sistemin işlemcisinde geliştirilabilecek bir makine kodu yığımına çevirir.





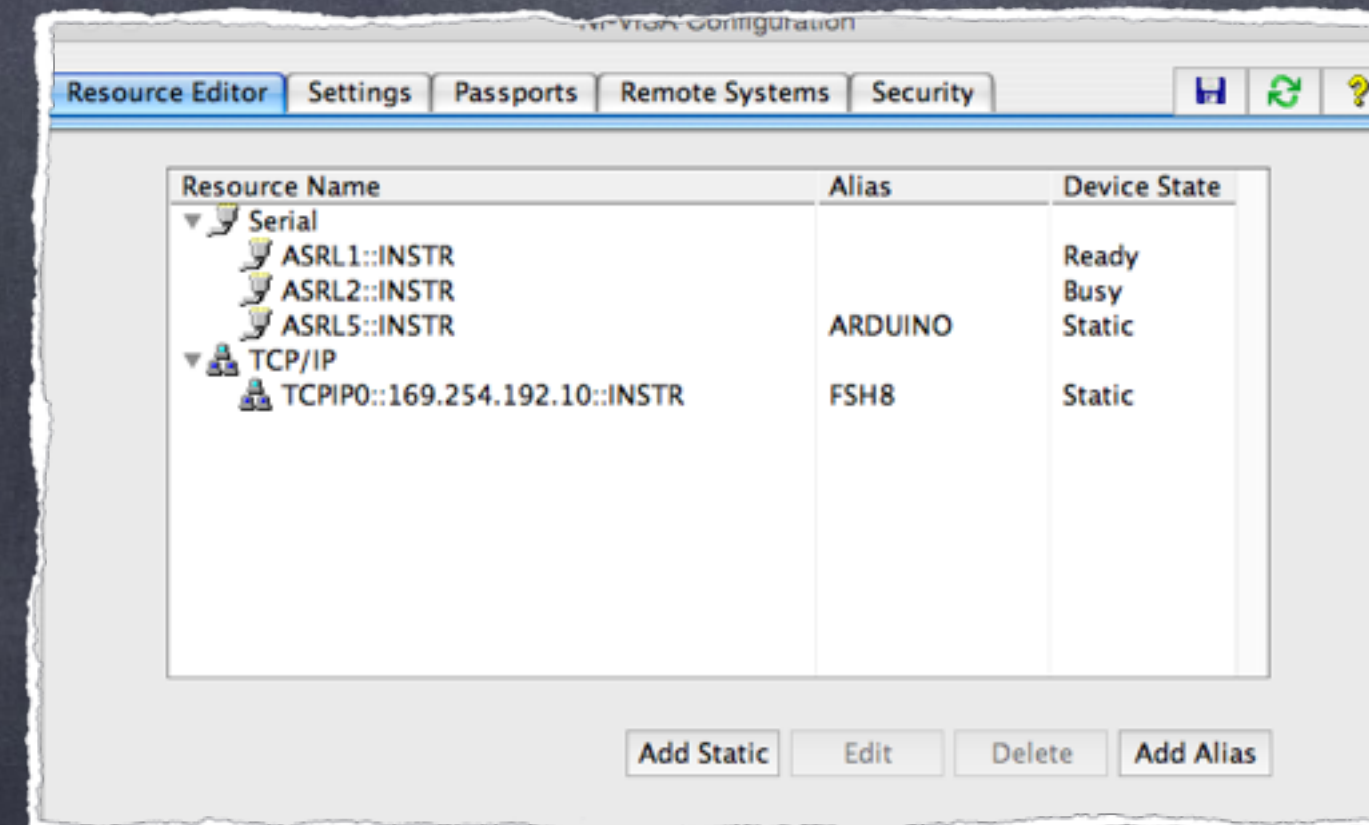
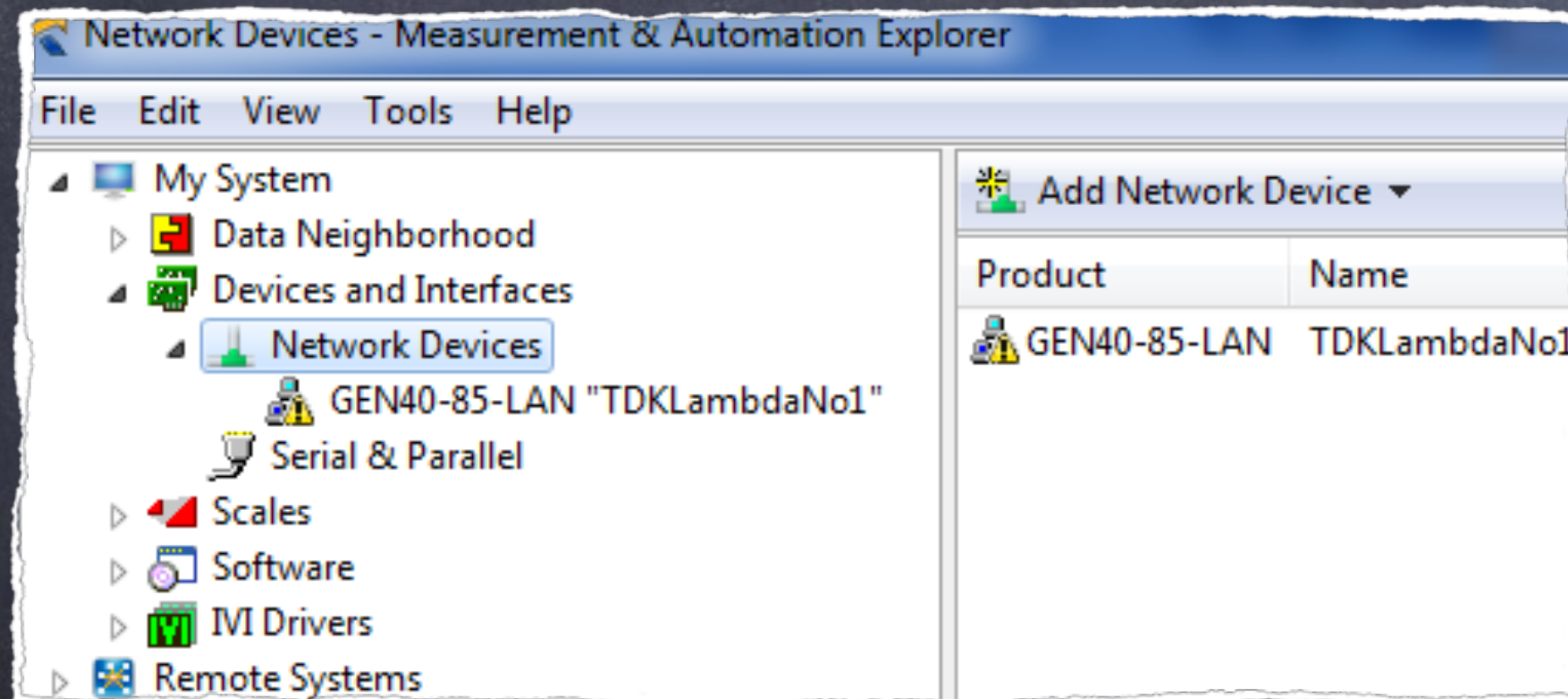
# Bilgisayar $\leftrightarrow$ Donanım

## -MAX-

NATIONAL INSTRUMENTS

### Measurement & Automation Explorer

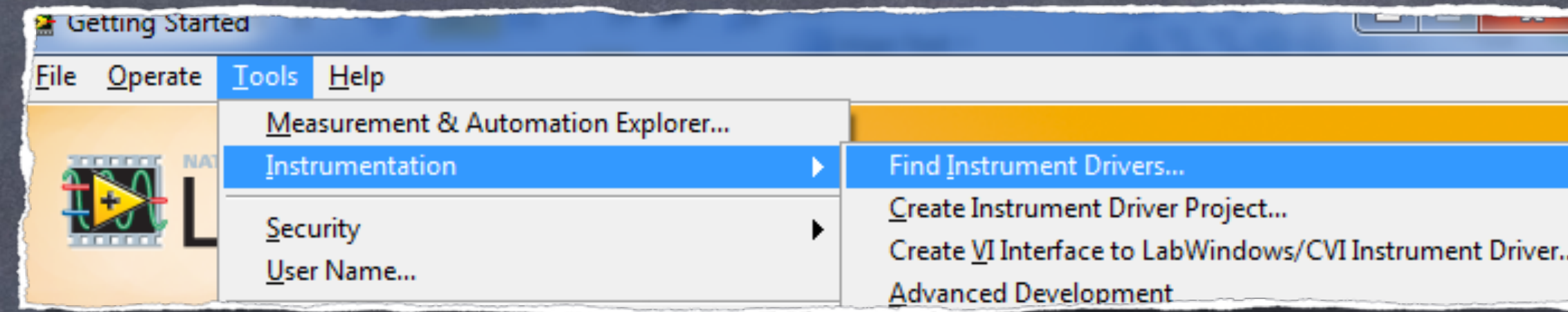
- NI DAQ, GPIB, IMAQ, IVI, PCI, Motion, VISA ve VXI bağlantı yollarına sahip donanımları LabVIEW'da test edebilme, kullanabilme ve düzenleyebilme imkanı sağlar.





# Donanım Sürücülerini

-NI-



The screenshot shows the 'Find Instrument Drivers - Search Results' window. It displays a table of drivers and their details. The table has columns for Driver, Driver Technology, NI Certified, and Rating. The 'Driver for LabVIEW 2012 SP1' is highlighted. To the right of the table, there are details for the selected driver, including a 4.0 rating, supported ADEs, required support software, driver revision, manufacturer, and supported models.

Driver	Driver Technology	NI Certified	Rating
GENie Instrument Driver	PnP	Yes	4,00
Driver for LabVIEW 2012 SP1			
Driver for LabVIEW 8.0			
Driver for LabVIEW 2010			
GENser Instrument Driver	PnP	Yes	3,70
genusb Instrument Driver	PnP Proj	Yes	NR

**4.0** ★★★★★ 4 ratings

**Driver ADE(s):**  
LabVIEW  
Min Version - 2012 SP1

**Required Support Software:**  
NI-VISA  
Min Version - 3.0

**Driver Revision:**  
1.0

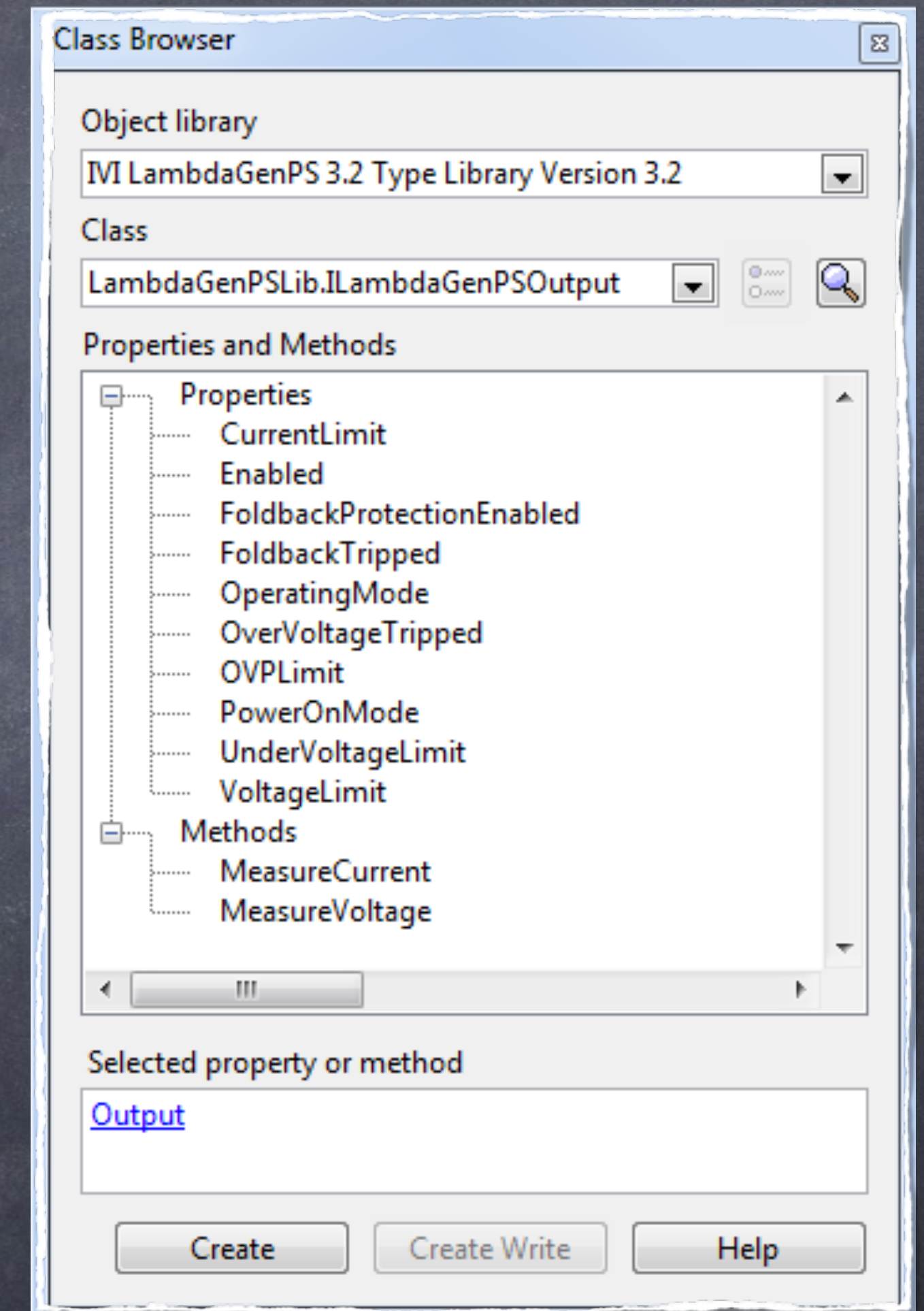
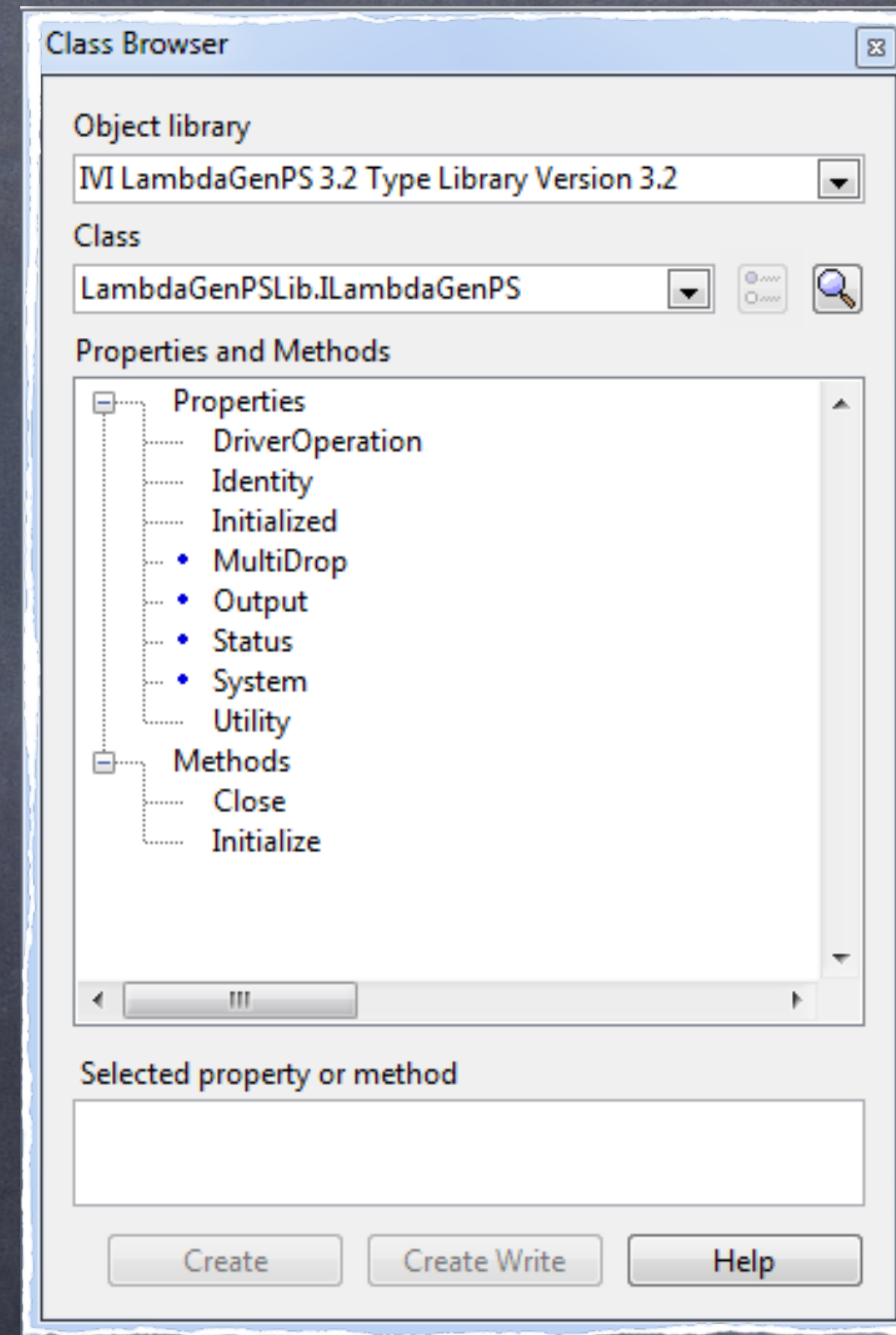
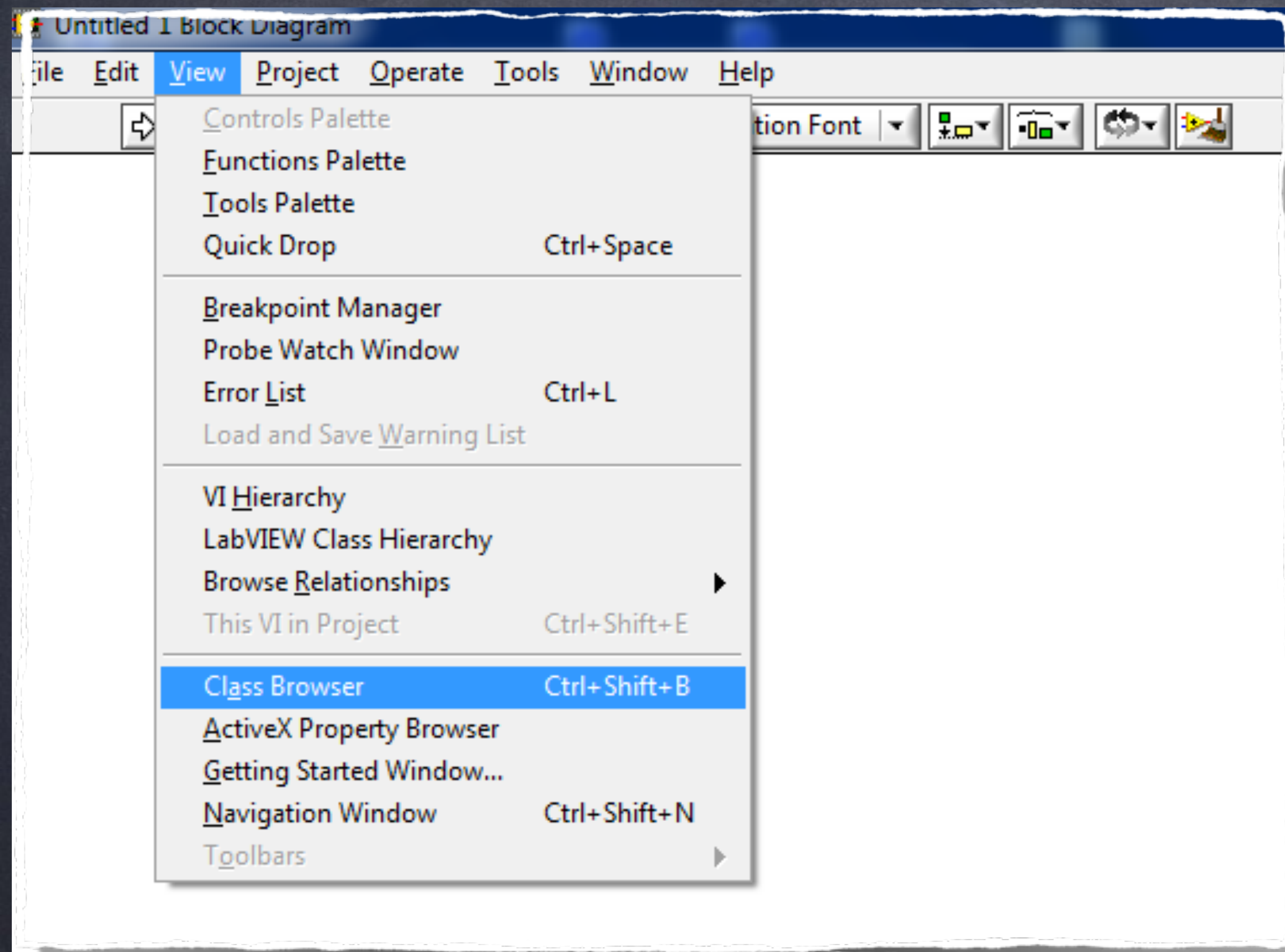
**Manufacturer:**  
Lambda

**Model(s) Supported:**  
Genesys (IEEE-488)

**Interface(s):**  
IEEE 488.2 (GPIB)



# Donanım Sürücülerini -IVI-COM-



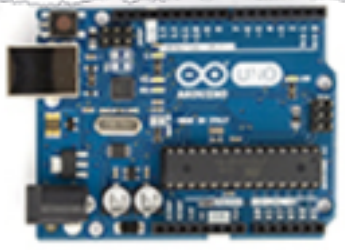


# Arduino Nedir?

- Açık kaynak fiziksel bir programlama altyapısıdır.
- Basitçe: Bilgisayar → donanım/bileşen arasında giriş/çıkış kartı.
- Kendi geliştirme ortamına (IDE) sahiptir.
- Mikro işlemci değildir!
  - Mikro işlemciler için kolaylaştırıcı bir geliştirme ortamı sağlar.
    - USB bağlantısı
    - Entegre LED
    - Giriş/çıkış bacakları
    - Güç giriş/çıkışı
    - vb...



# Arduinolar



Arduino Uno



Arduino Due



Arduino Tre



Arduino Micro



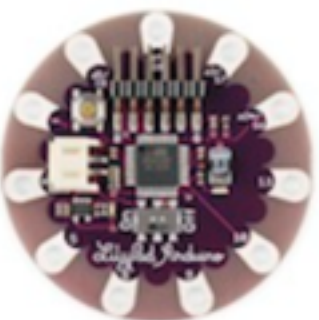
Arduino Mega ADK



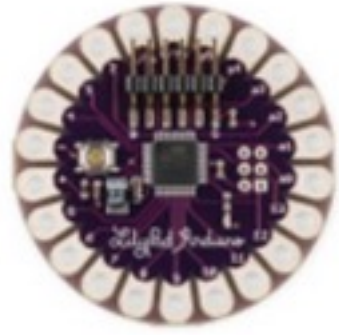
Arduino Mega 2560



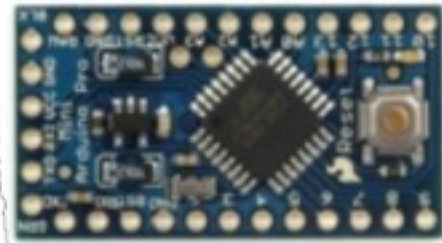
Arduino Mini



LilyPad Arduino Simple



LilyPad Arduino



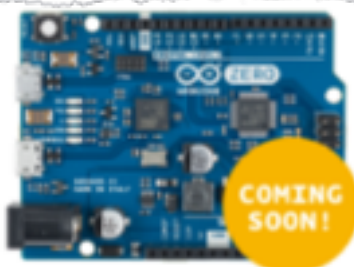
Arduino Pro Mini



Arduino Pro



Arduino Leonardo



Arduino Zero



Arduino Esplora



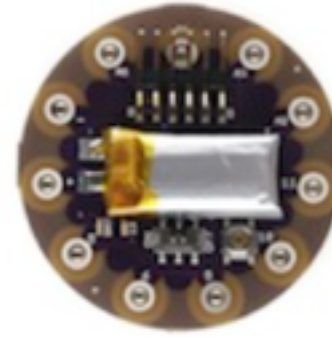
Arduino Ethernet



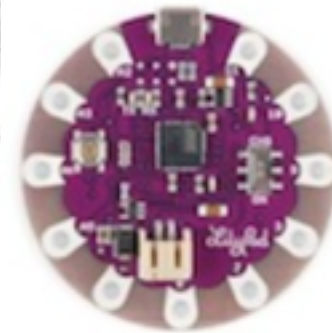
Arduino Robot



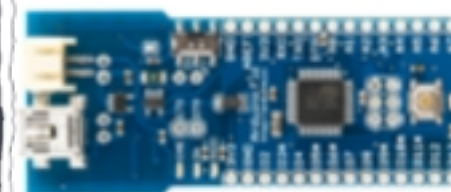
Arduino Nano



LilyPad Arduino SimpleSnap



LilyPad Arduino USB



Arduino Fio



# Eklentiler



Arduino GSM Shield



Arduino Ethernet Shield



Arduino WiFi Shield



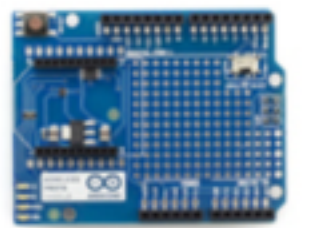
Arduino Wireless SD Shield



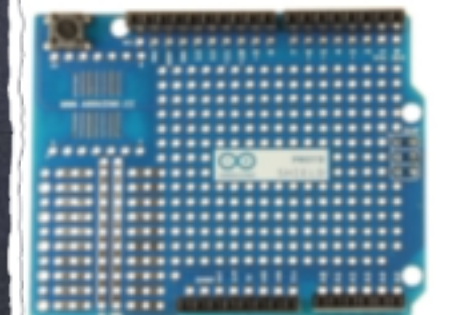
Arduino USB Host Shield



Arduino Motor Shield



Arduino Wireless Proto Shield



Arduino Proto Shield



# Farklar

<http://arduino.cc/en/Products.Compare>

Name	Processor	Operating Voltage/Input Voltage	CPU Speed	Analog In/Out	Digital IO/PWM	EEPROM [KB]	SRAM [KB]	Flash [KB]	USB	UART
Uno	ATmega328	5 V/7-12 V	16MHz	6/0	14/6	1	2	32	Regular	1
Due	AT91SAM3X8E	3.3 V/7-12 V	84 MHz	12/2	54/12	-	96	512	2 Micro	4
Leonardo	ATmega32u4	5 V/7-12 V	16MHz	12/0	20/7	1	2.5	32	Micro	1
Mega 2560	ATmega2560	5 V/7-12 V	16MHz	16/0	54/15	4	8	256	Regular	4
Mega ADK	ATmega2560	5 V/7-12 V	16MHz	16/0	54/15	4	8	256	Regular	4
Micro	ATmega32u4	5 V/7-12 V	16MHz	12/0	20/7	1	2.5	32	Micro	1
Mini	ATmega328	5 V/7-9 V	16MHz	8/0	14/6	1	2	32	-	-
Nano	ATmega168	5 V/7-9 V	16MHz	8/0	14/6	0.512	1	16	Mini-B	1
	ATmega328					1	2	32		
Ethernet	ATmega328	5 V/7-12 V	16MHz	6/0	14/4	1	2	32	Regular	-
Esplora	ATmega32u4	5 V/7-12 V	16MHz	-	-	1	2.5	32	Micro	-
ArduinoBT	ATmega328	5 V/2.5-12 V	16MHz	6/0	14/6	1	2	32	-	1
Fio	ATmega328P	3.3 V/3.7-7 V	8MHz	8/0	14/6	1	2	32	Mini	1
Pro (168)	ATmega168	3.3 V/3.35-12 V	8MHz	6/0	14/6	0.512	1	16	-	1
Pro (328)	ATmega328	5 V/5-12 V	16MHz	6/0	14/6	1	2	32	-	1
Pro Mini	ATmega168	3.3 V/3.35-12 V	8MHz	6/0	14/6	0.512	1	16	-	1
		5 V/5-12 V	16MHz							



Analog referans gerilim giriři Dijital toprak Dijital giriř/řıkıř bacakları

Serial řıkıř  
Serial giriř

Sıfırlama tuřu

USB giriři

PWM? řıkıřlar

ICSP baęlantısı

ATmega328  
mikrokontrolcü

Harici  
güç kaynaęı  
giriři

Analog giriř  
bacakları

Giriř/řıkıř gerilim  
referansı

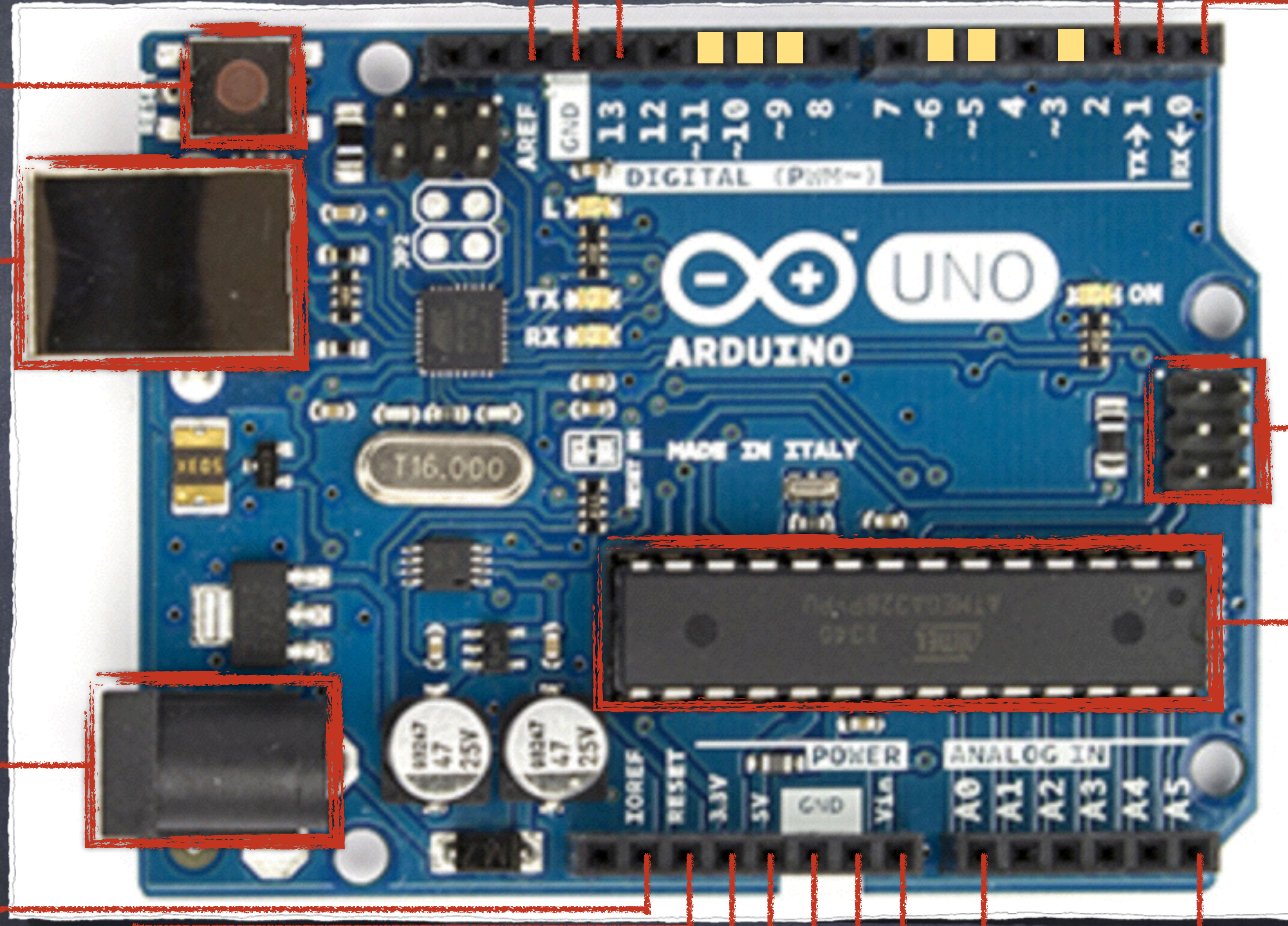
Sıfırlama  
giriři

3.3V  
řıkıř

5V řıkıř

Toprak  
giriři

Voltaj giriři



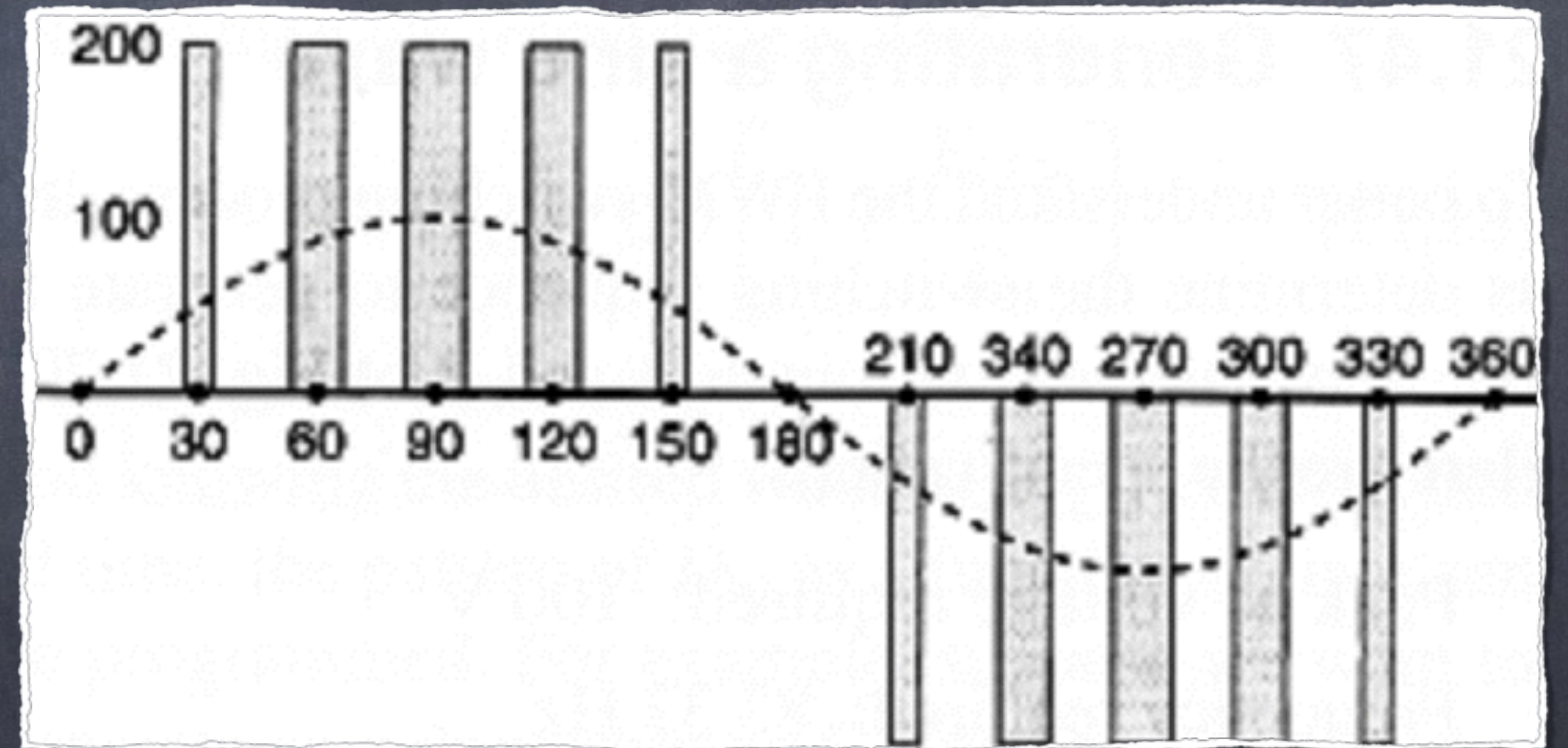
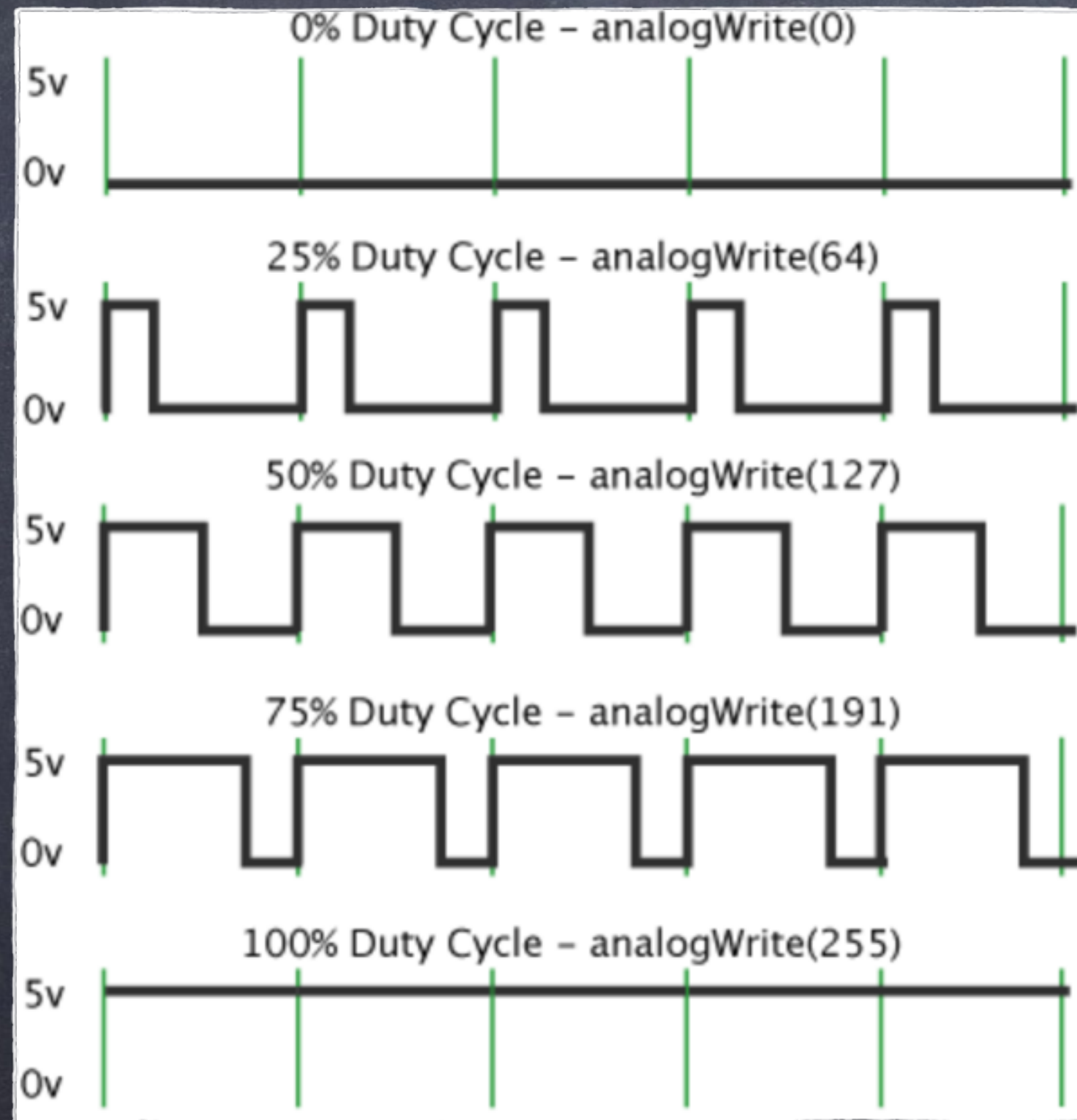


# PWM (Sinyal genişlik kiplenimi) Nedir?

-Dijital sinyaller ile "çakma" analog sinyal-

Duty Cycle: Bir sinyalin zamanın % kaçında "yüksek" olduğu = atma genişliği/dönüm

Arduino'da PWM → `analogWrite(pin, duty cycle)`






# Ortam ve Dil

<http://arduino.cc/en/Main/Software>

<http://arduino.cc/en/Reference/HomePage>



**ARDUINO 1.0.6**  
The open-source Arduino Software (IDE) makes it easy to write code and upload it to the board. It runs on Windows, Mac OS X, and Linux. The environment is written in Java and based on Processing and other open-source software.  
This software can be used with any Arduino board. Refer to the [Getting Started](#) page for Installation instructions.

**Windows** Installer  
**Windows** ZIP file for non admin instal

**Mac OS X**

**Linux** 32 bits  
**Linux** 64 bits

[Release Notes](#)

```
stepperdrv8825 | Arduino 1.0.6  
stepperdrv8825  
int dirPin = 11;  
int stepperPin = 12;  
  
void setup() {  
  pinMode(dirPin, OUTPUT);  
  pinMode(stepperPin, OUTPUT);  
  digitalWrite(13, HIGH);  
}  
  
void step(boolean dir,int steps){  
  digitalWrite(dirPin,dir);  
  delay(50);  
  for(int i=0;i<steps;i++){  
    digitalWrite(stepperPin, HIGH);  
    delayMicroseconds(800);  
    digitalWrite(stepperPin, LOW);  
    delayMicroseconds(800);  
  }  
}  
  
void loop(){  
  step(true,800);  
  delay(500);  
  step(false,800);  
  delay(500);  
}  
  
Done Saving.  
24 Arduino Uno on /dev/tty.usbmodem1421
```

## Structure

- setup()
- loop()

## Control Structures

- if
- if...else
- for
- switch case
- while
- do... while
- break
- continue
- return
- goto

## Further Syntax

- ; (semicolon)
- {} (curly braces)
- // (single line comment)
- /\* \*/ (multi-line comment)
- #define
- #include

## Variables

### Constants

- HIGH | LOW
- INPUT | OUTPUT | INPUT\_PULLUP
- LED\_BUILTIN
- true | false
- integer constants
- floating point constants

### Data Types

- void
- boolean
- char
- unsigned char
- byte
- int
- unsigned int
- word
- long
- unsigned long
- short
- float
- double

## Functions

### Digital I/O

- pinMode()
- digitalWrite()
- digitalRead()

### Analog I/O

- analogReference()
- analogRead()
- analogWrite() - PWM

### Due only

- analogReadResolution()
- analogWriteResolution()

### Advanced I/O

- tone()
- noTone()
- shiftOut()
- shiftIn()
- pulseIn()

### Time

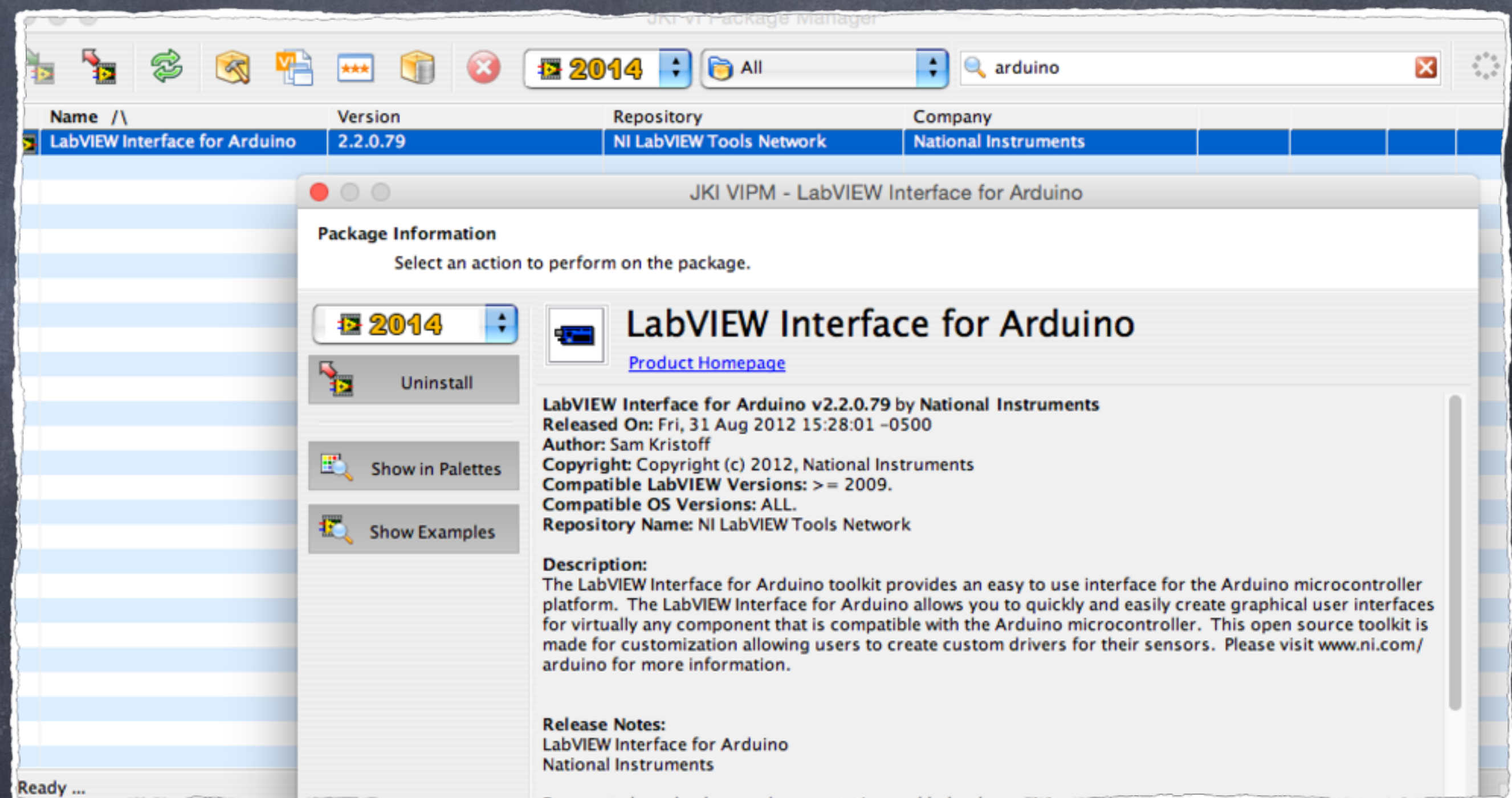
millis()



# Arduino @ LabVIEW

- Arduino IDE
- NI-VISA drivers
- VI-package manager
- Labview interface for arduino

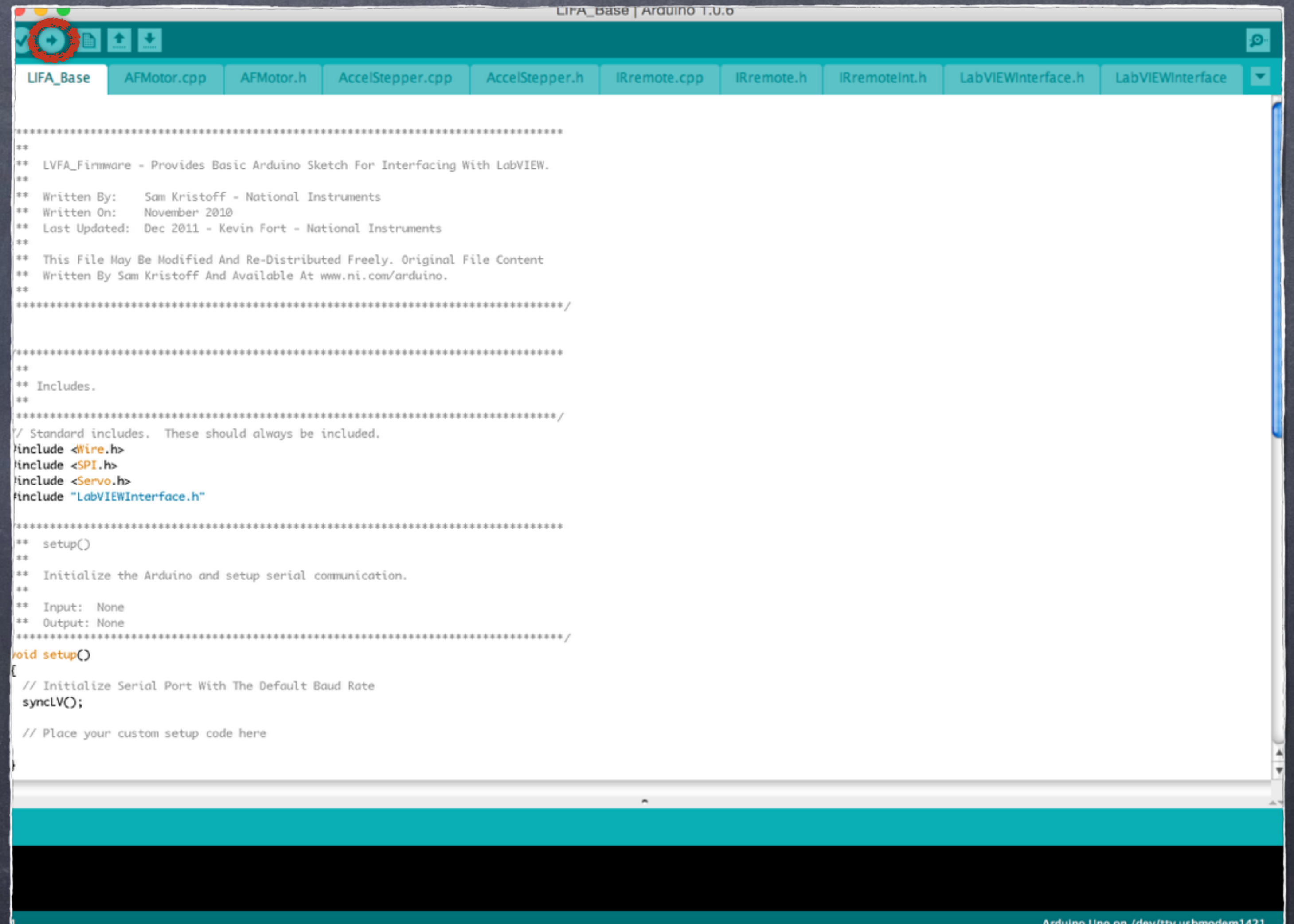
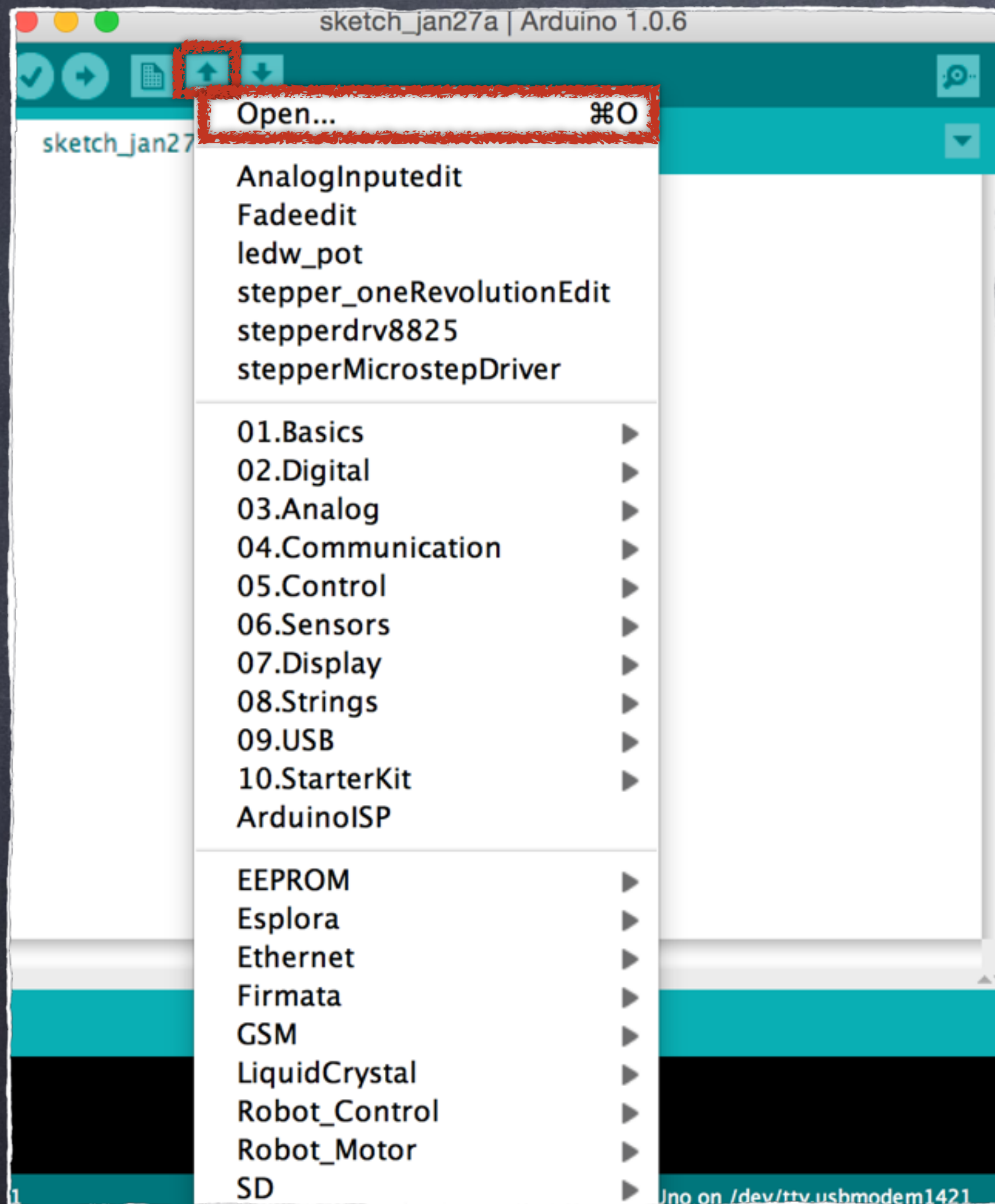
Arduino IDE   NI-VISA sürücüleri   VI-Paket Yöneticisi





# Arduino <-> LabVIEW Tanıstırma

National Instruments > LabVIEW 2014 > vi.lib > LabVIEW Interface for Arduino > Firmware > LIFA\_Base > LIFA\_Base.ino



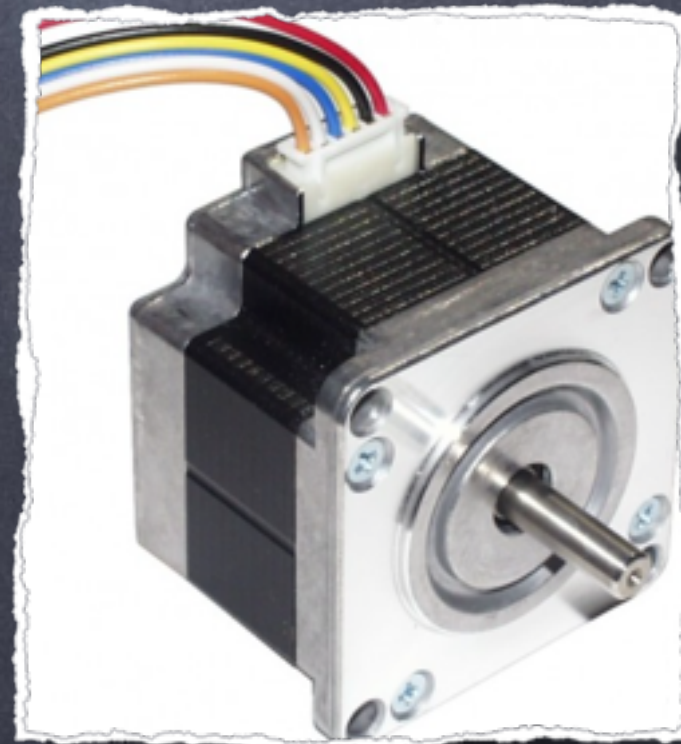
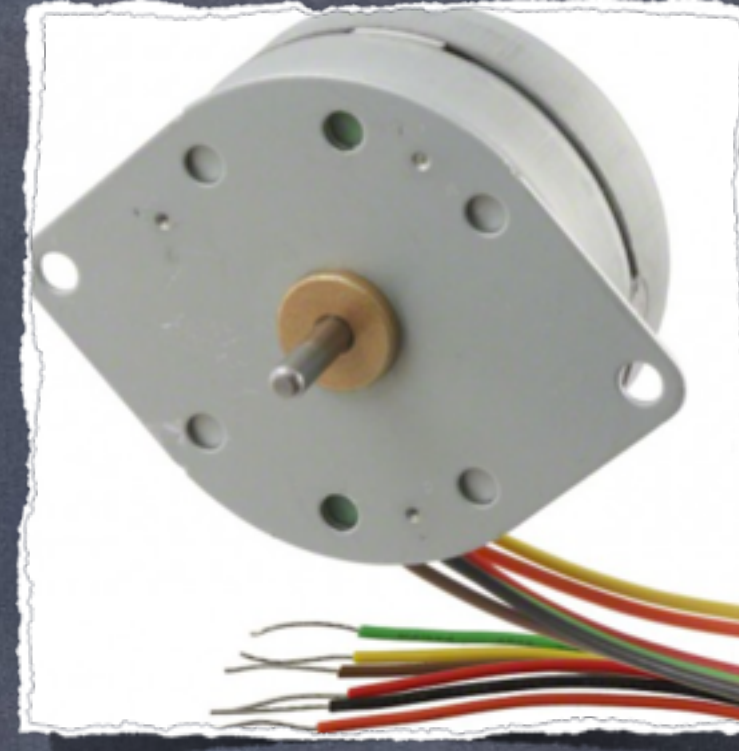


# Motorlar

Doğru akım



Adım



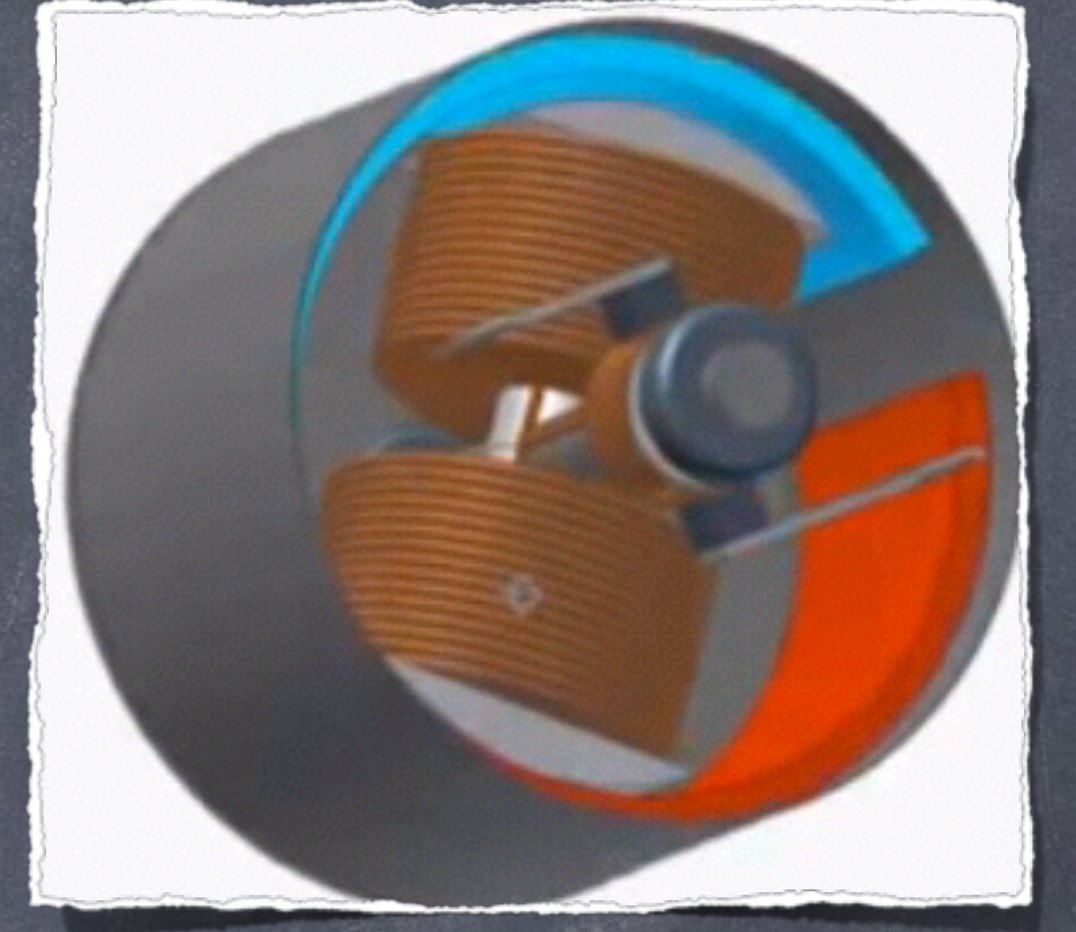
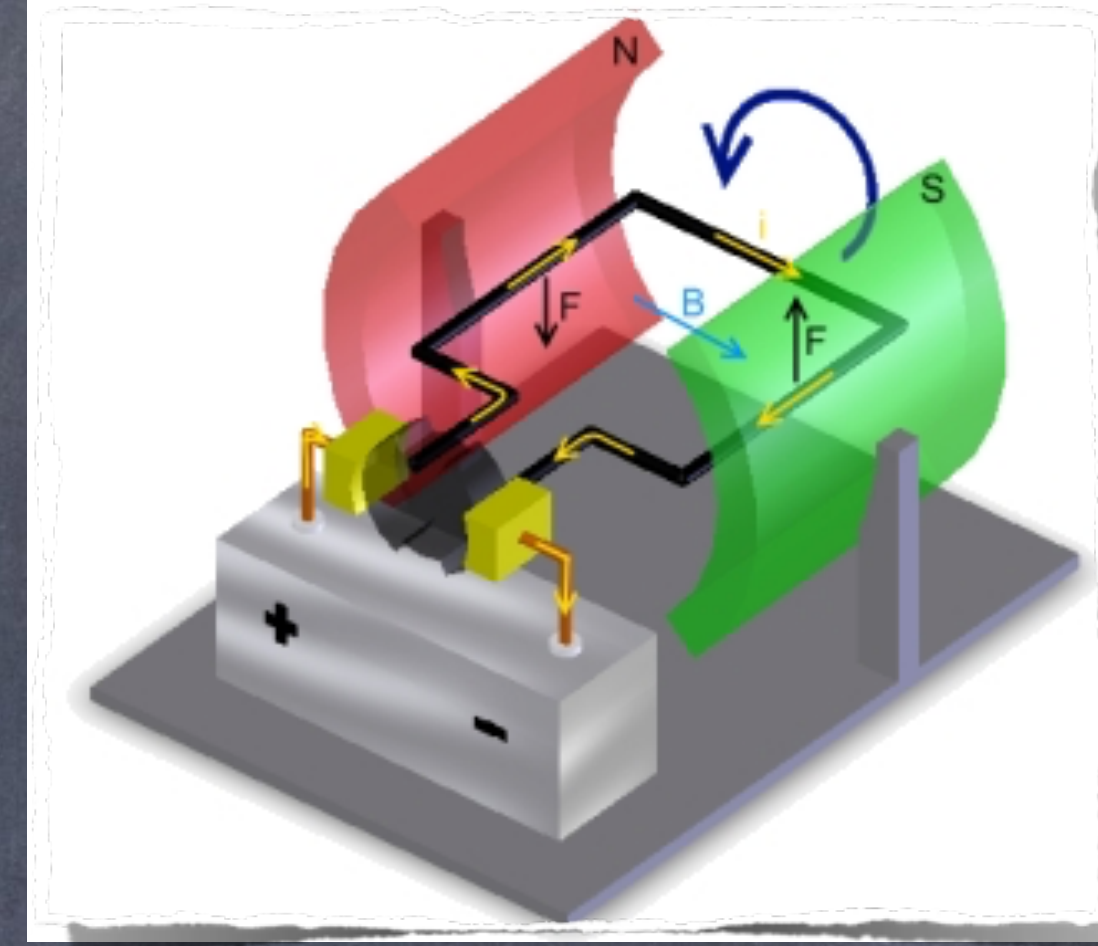
Açık





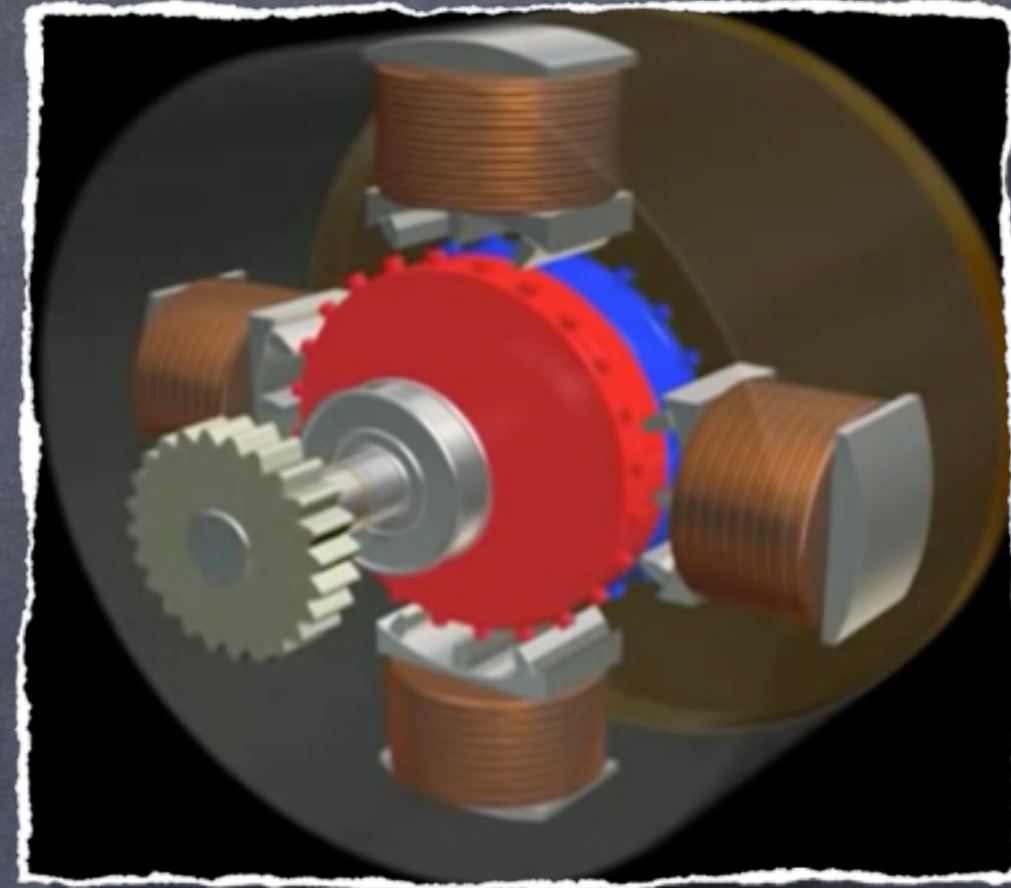
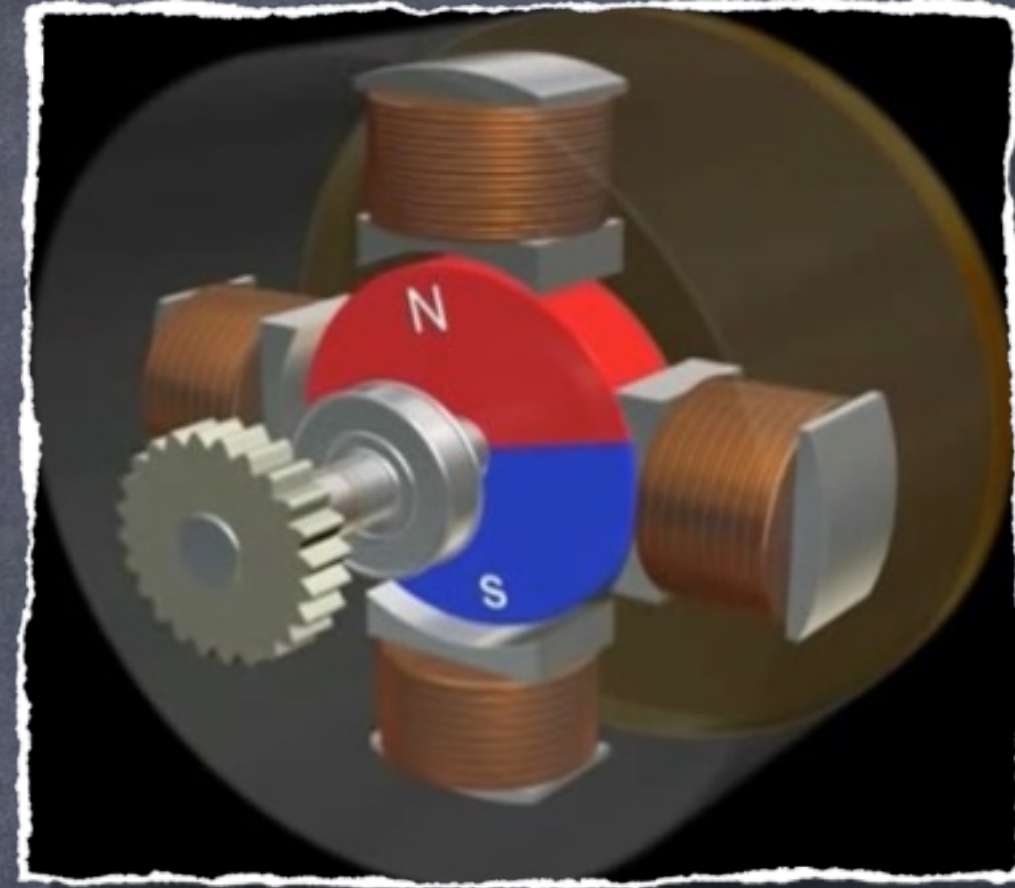
# Doğru Akım Motorlar

- 2 kablosu vardır.
- Güç (kırmızı)
- Toprak (siyah)
- Güç verildiği sürece motor dönmeye devam eder.
- Motor dönüş hızı PWM ile ayarlanır.
- Bilgisayar pervaneleri, uzaktan kumandalı araba tekerlekleri, vb.





# Adım Motorlar

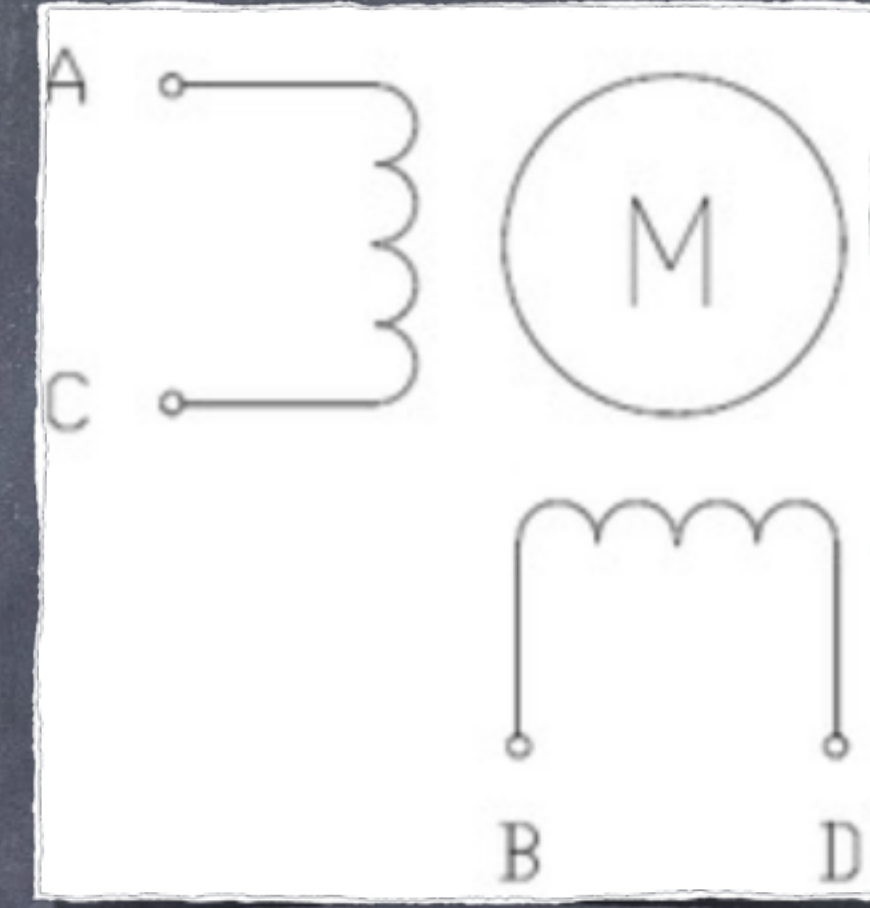


- Elektromagnatların sırayla beslenmesi prensibiyle motor döner.
- Harici bir kontrol devresi yada bir mikrokontrolcü gerektirir.
- Geometrisi nedeniyle belirli adım (mesafe) kadar hareket eder.
- 3B yazıcılar, tarayıcılar, vb.

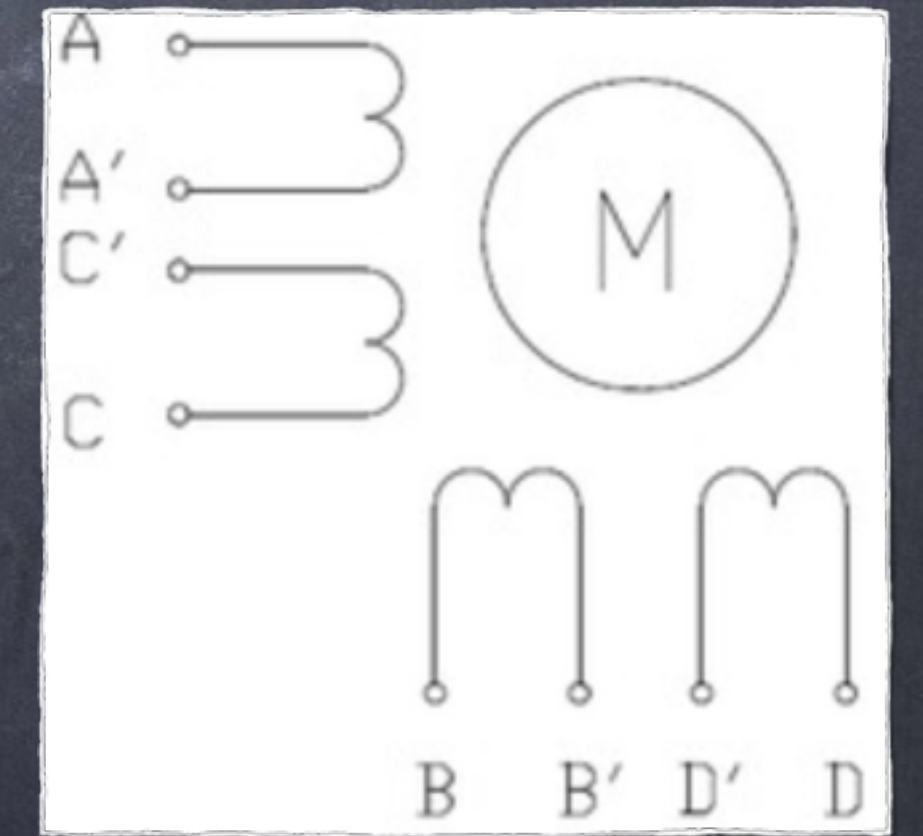
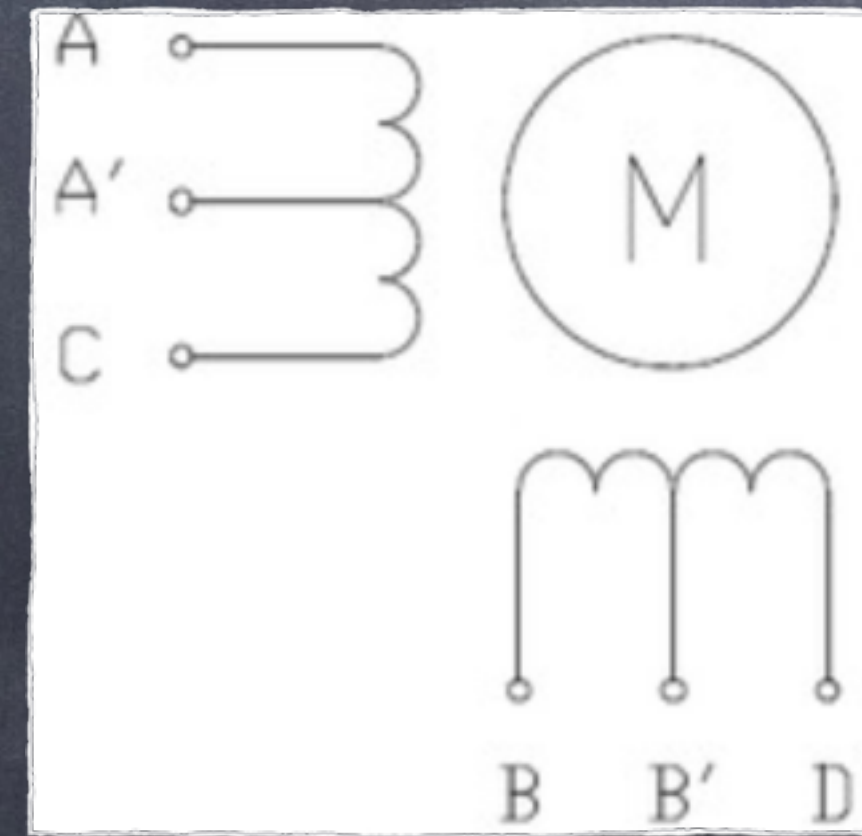


# Adım Motorlar

- İki uçaylı
  - 4,6 yada 8 kablosu vardır.
  - İki yönde de akım geçebilir.

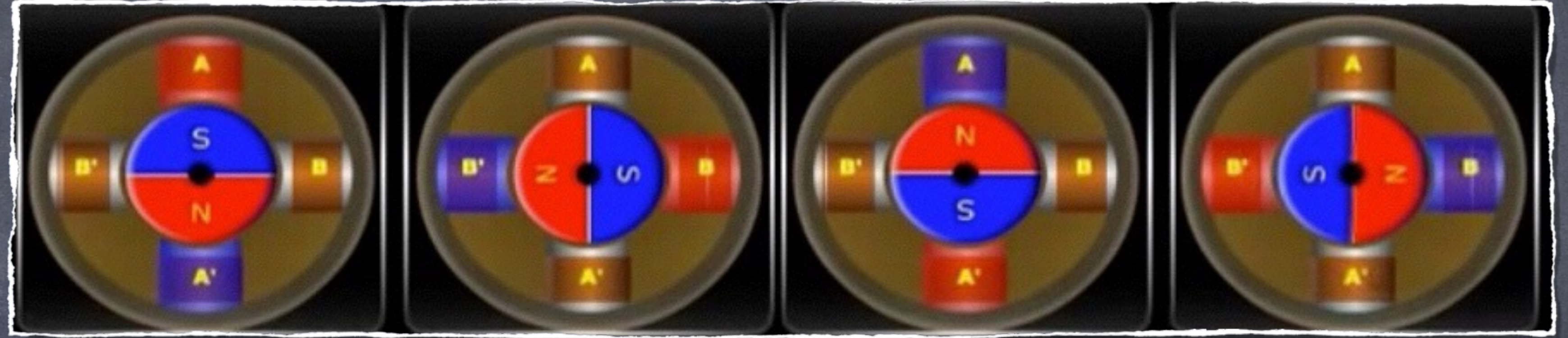
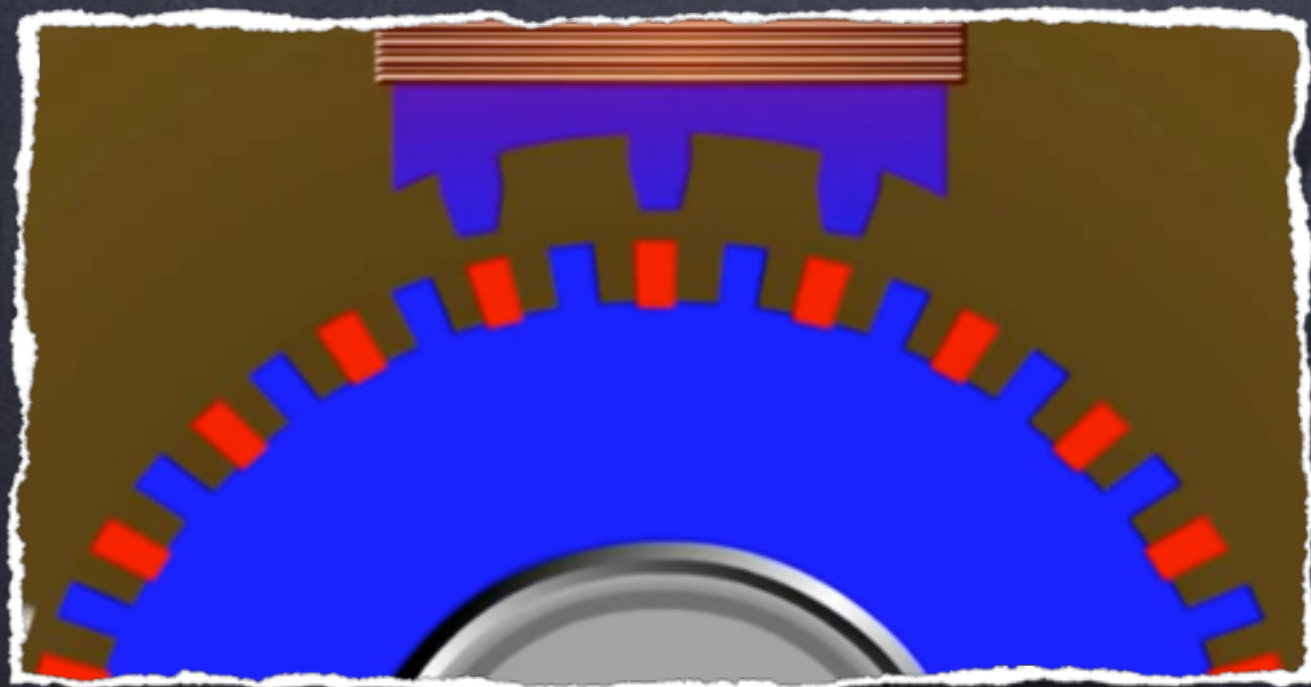
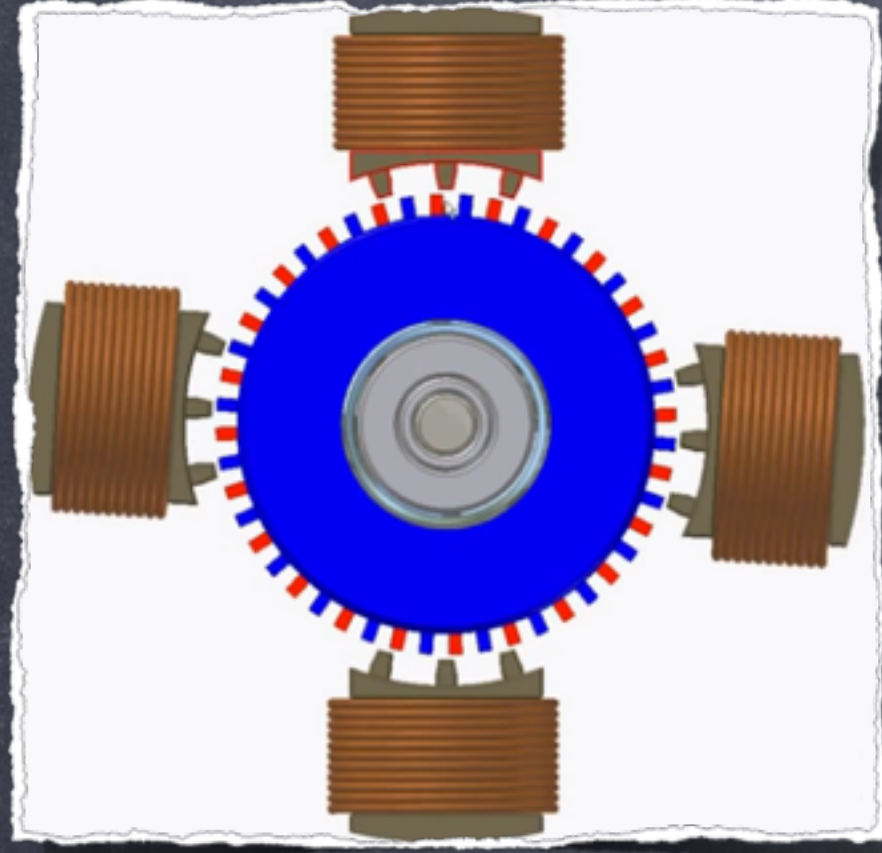
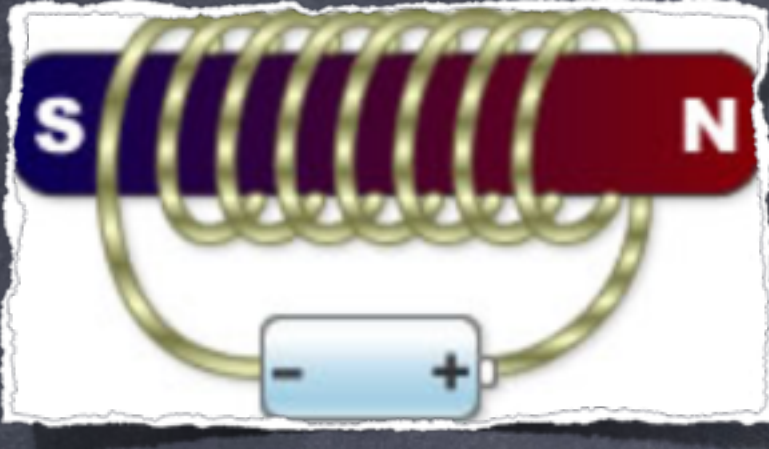
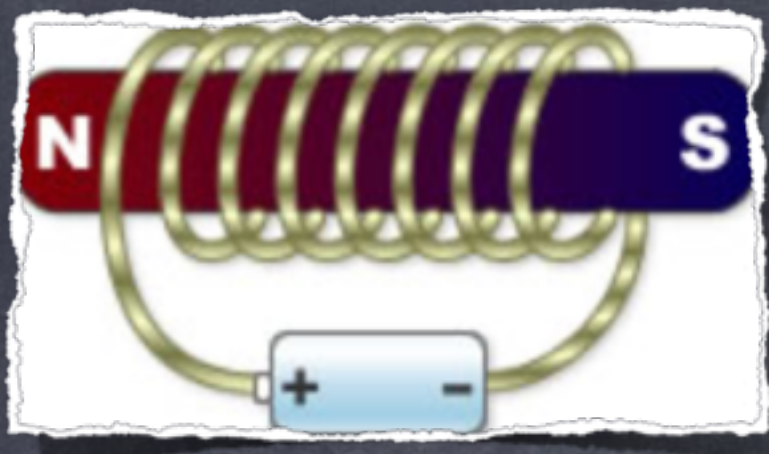


- Tek uçaylı
  - 5,6 yada 8 kablosu vardır.
  - Tek yönde akım geçer.

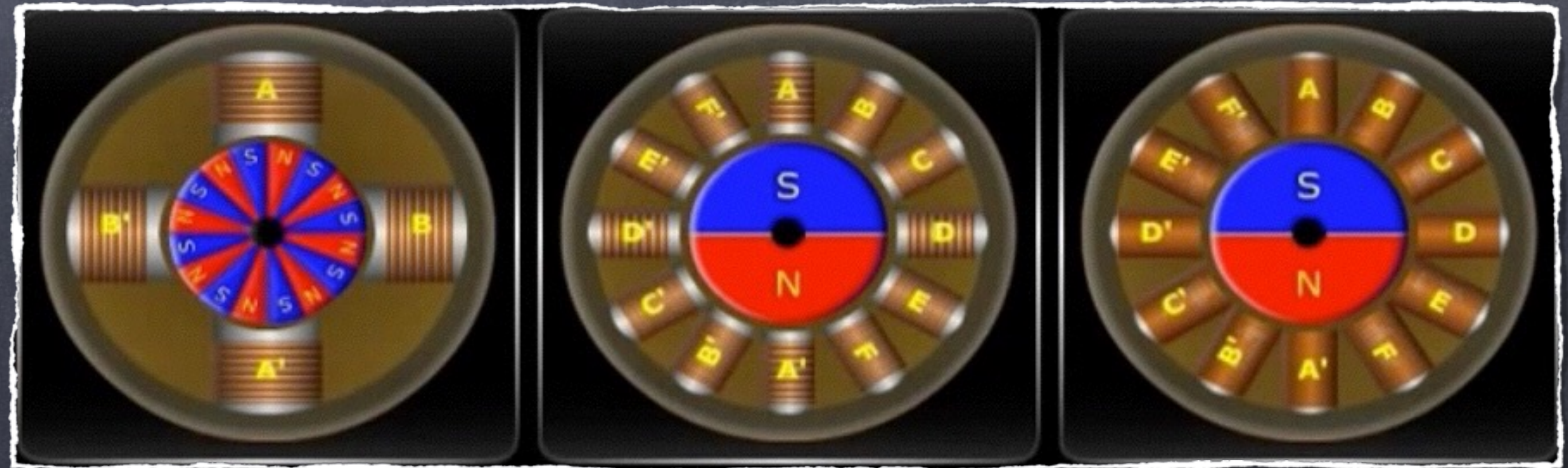




# Adım Motorlar



Hassaslığı arttırmak için yapılabilecekler:



Çarktaki uçay sayısı

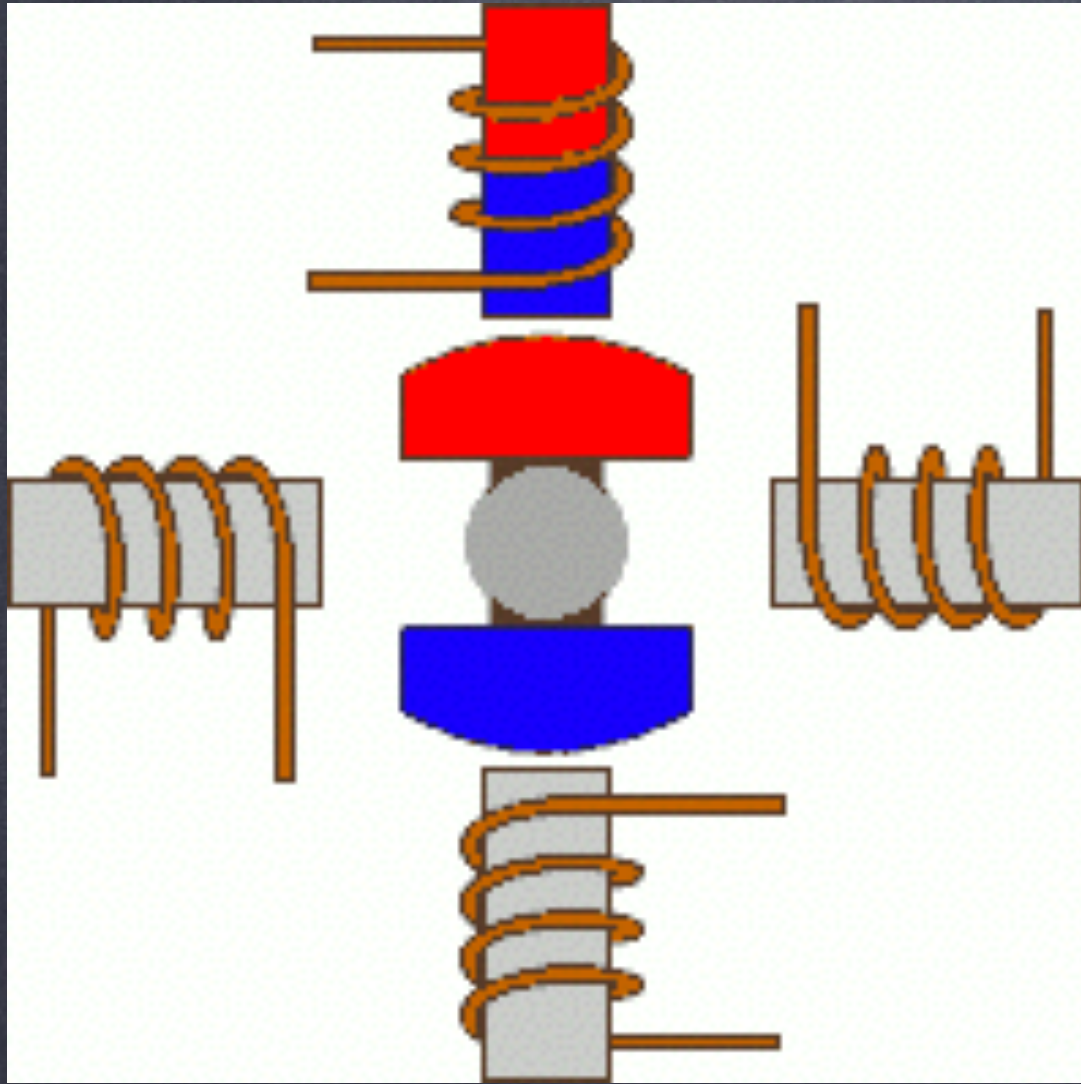
Fazlardaki uçay sayısı

Faz başına sarım sayısı



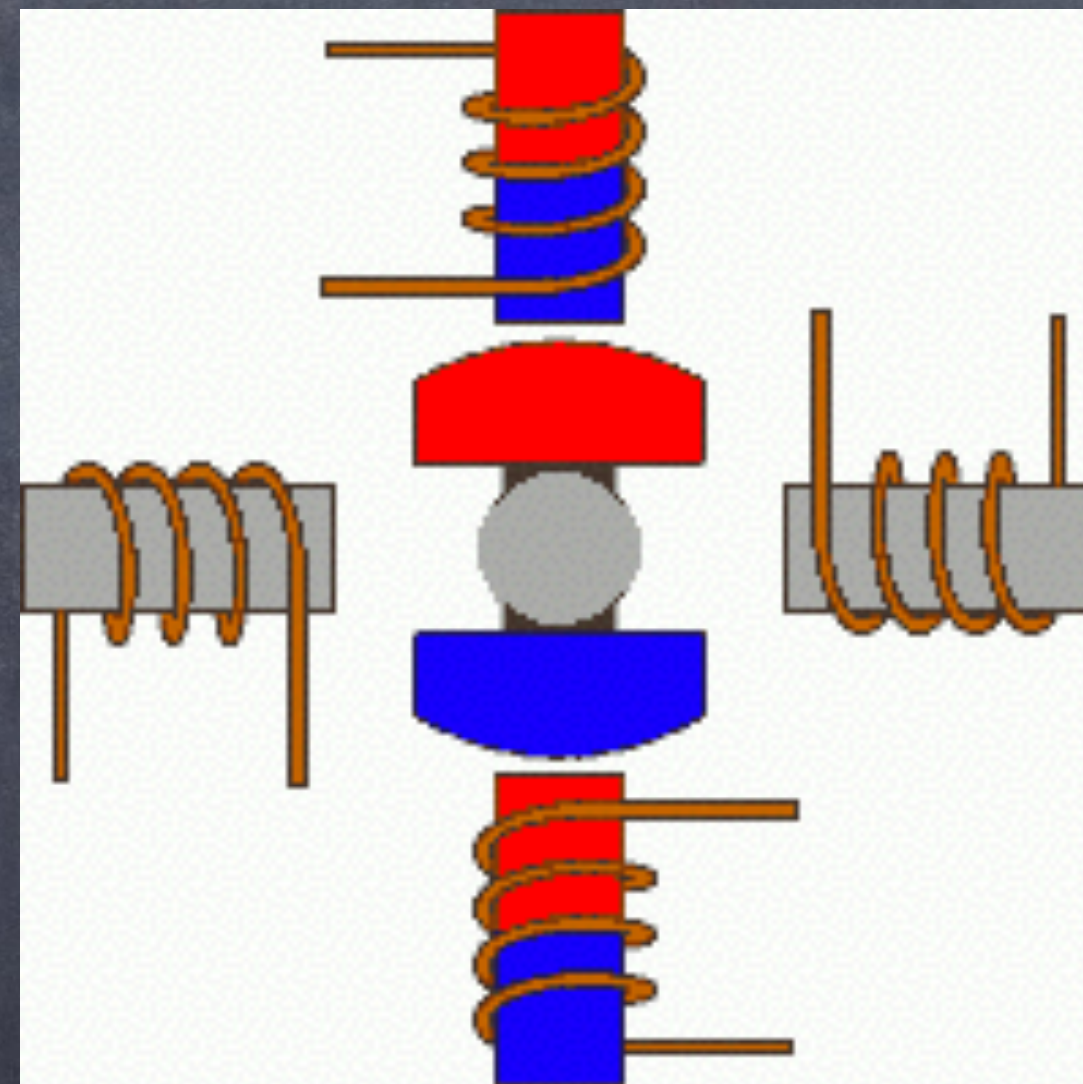
# Motor Sürme Kipleri

Dalga sürme



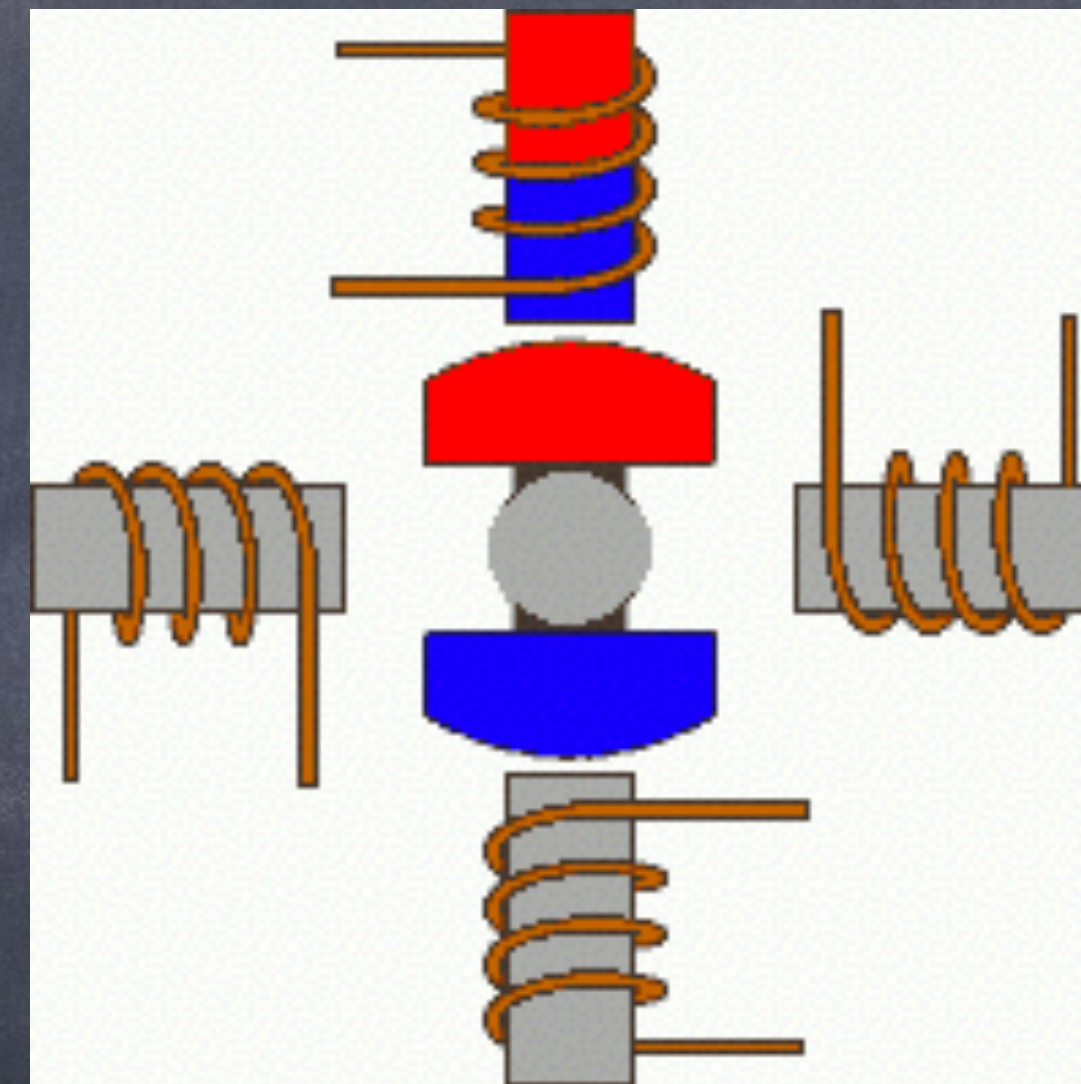
< % 50 dönme kuvveti

Tam adım sürme



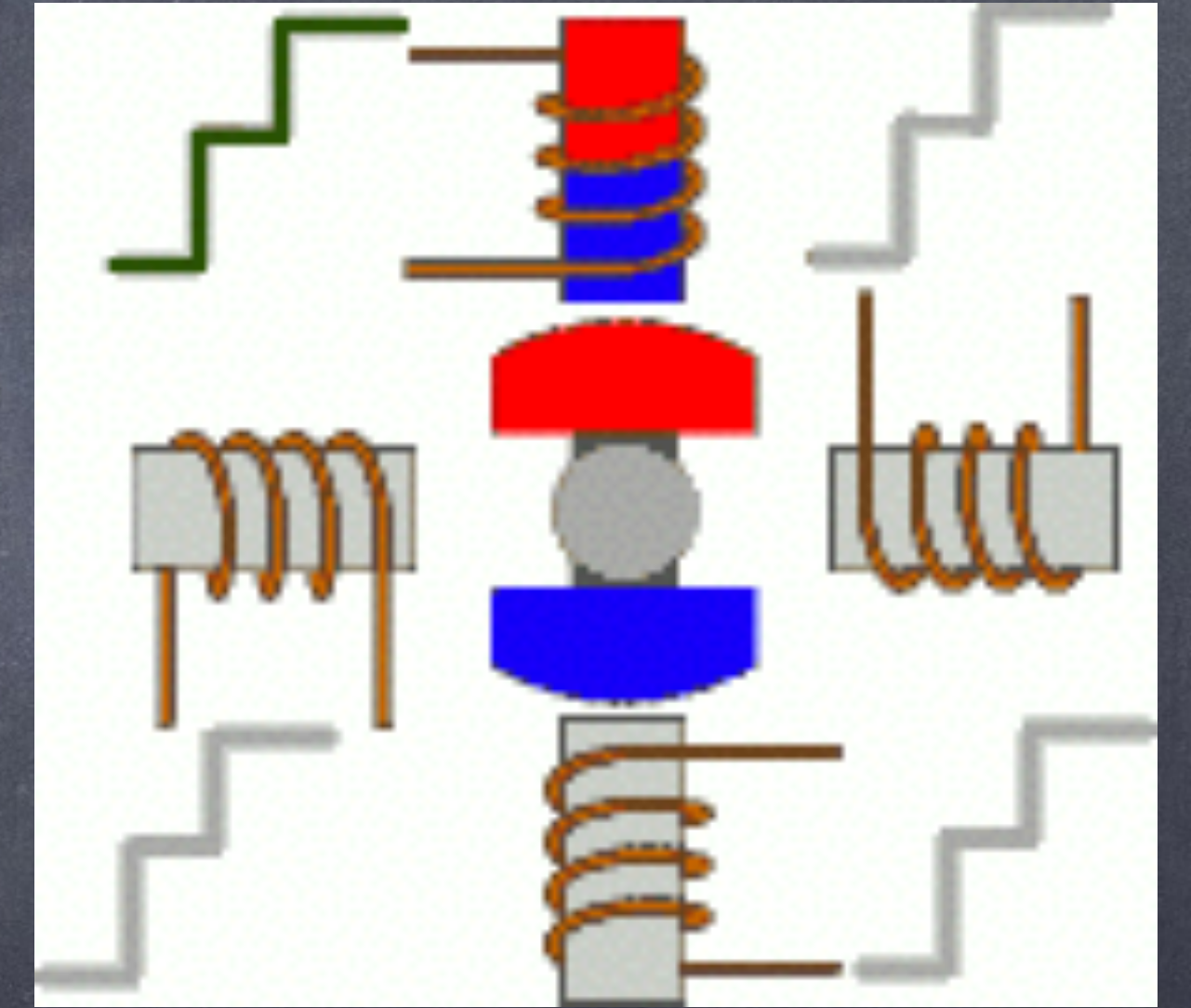
% 100 dönme kuvveti

Yarım adım sürme



x2 adım sayısı (hassaslık)

Mikroadım sürme



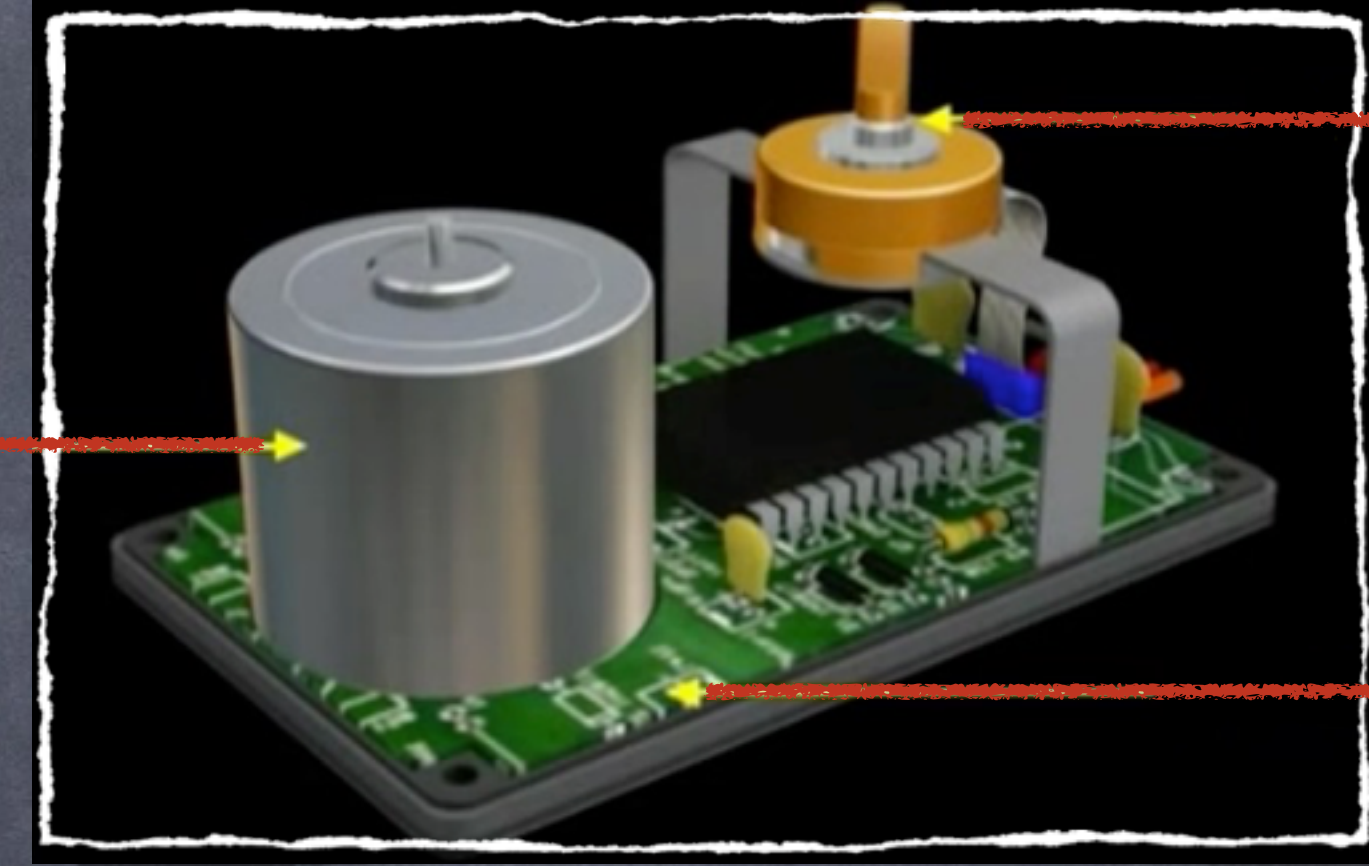
x4 / x8 / x16 / x32 / ...  
adımlar arasında  
yumuşak geçiş



# Ağı Motorlar

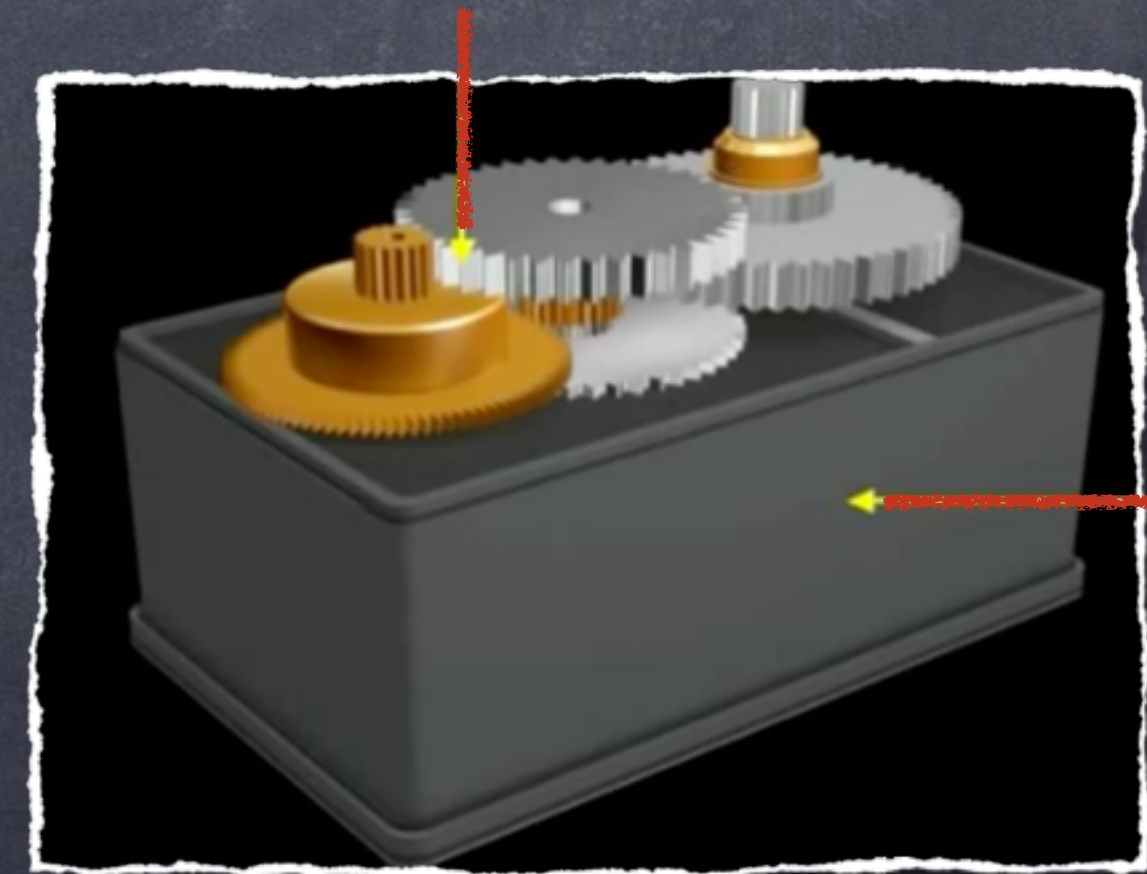
- 3 kablosu vardır.
  - Güç (kırmızı)
  - Toprak (kahverengi yada siyah)
  - Kontrol (beyaz, turuncu yada sarı)
- Yüksek akımlar gerektirmez.
- Güç sürekli sağlanır, kontrol devresi motor beslemesini ayarlar.
- Ayarlı direnç sayesinde motorun açısı kontrol devresi tarafından takip edilir.
- Motor açısı PWM ile ayarlanır.
- Robotik kollar, dümen sistemleri, vb.

doğru akım motoru



ayarlı direnç (konum algıcı) kontrol devresi

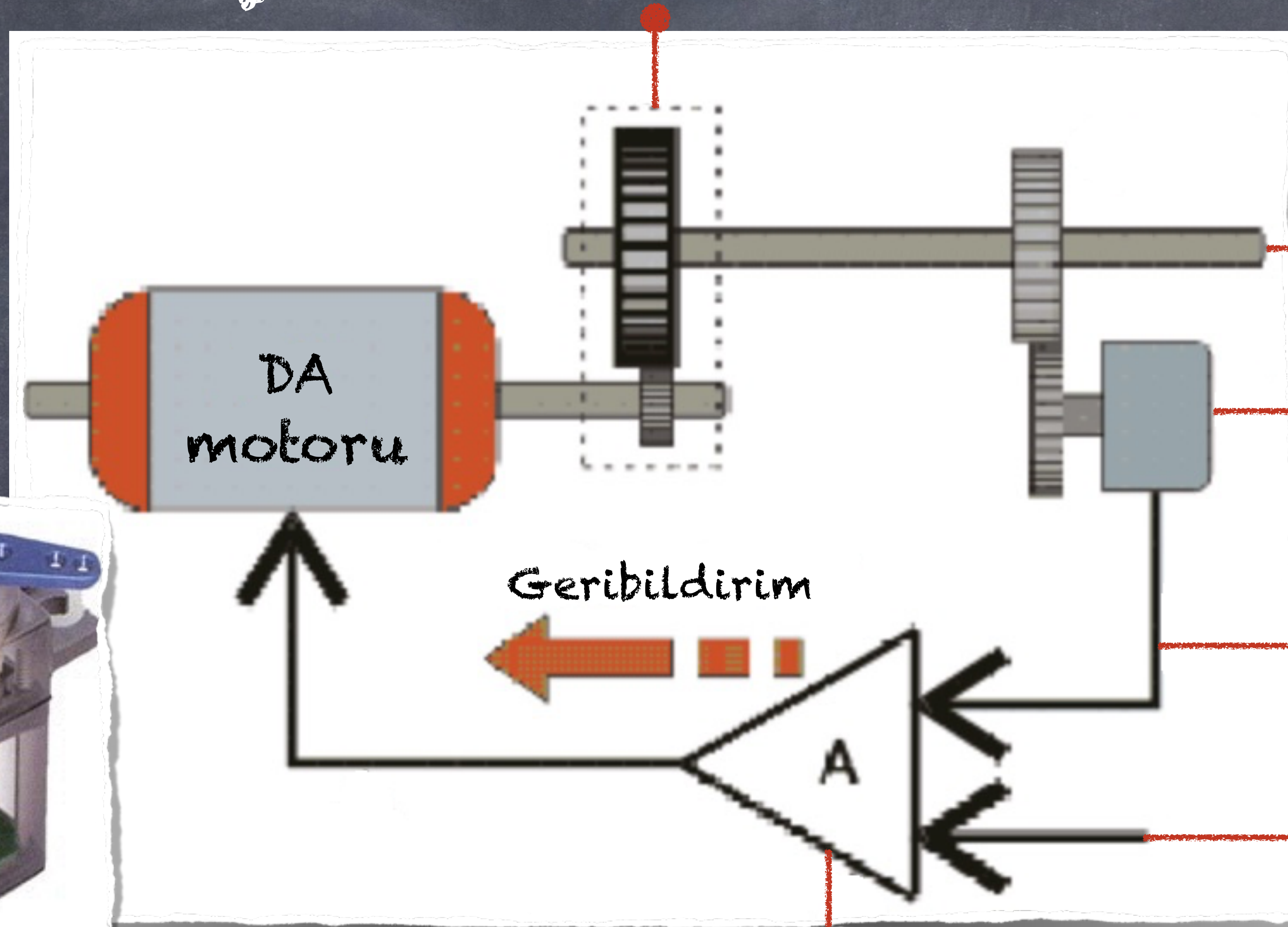
dişli takımı (dönme kuvvetini arttırmak için)



kılıf



# Açı Motor İlke



Çıkış mili

Ayarlı direnç

Referans çıkış sinyali

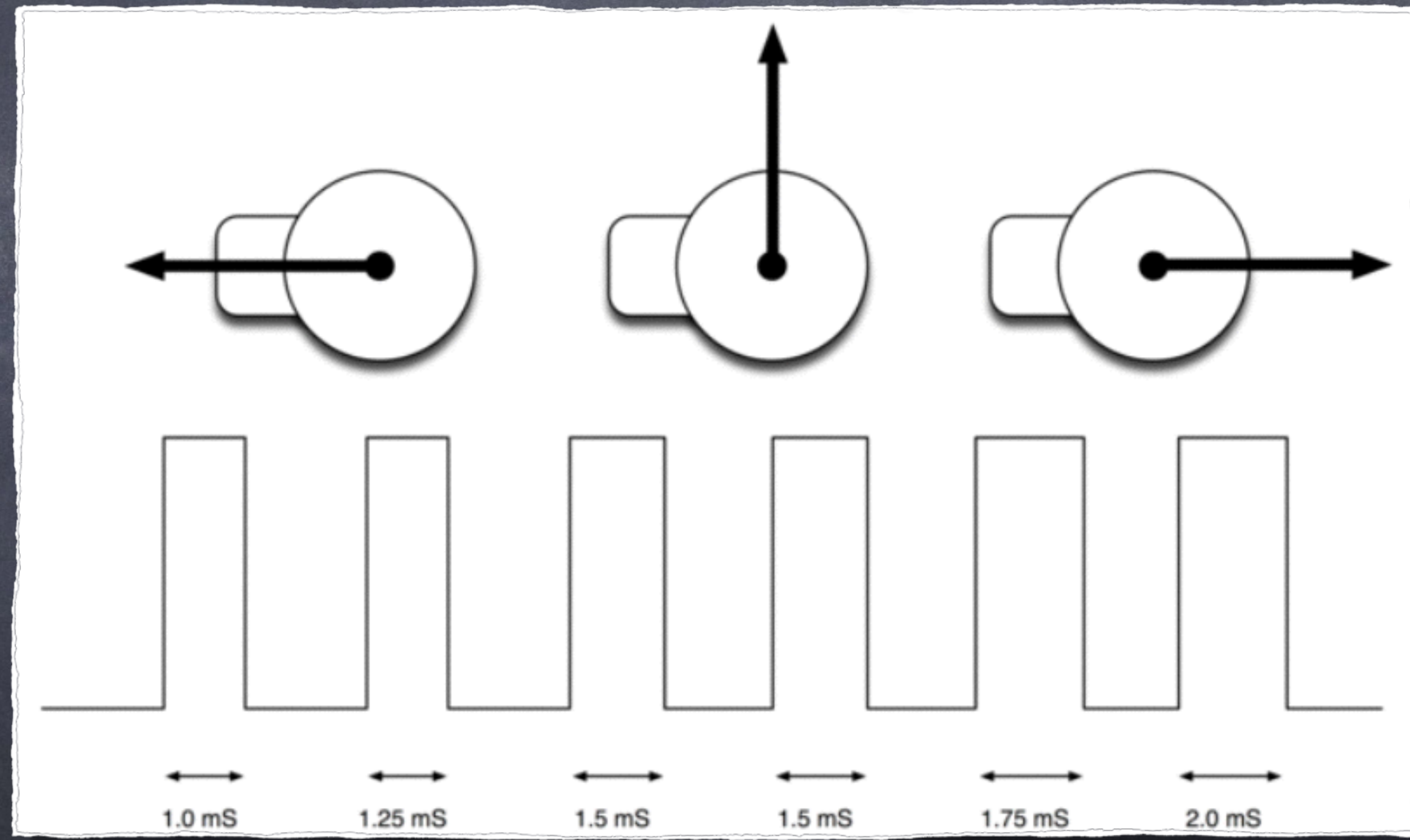
Referans giriş sinyali

Hata algılayıcı





# Ağı Motor Atma Uzunluğu





# Uygulama...

- Adımcılar:

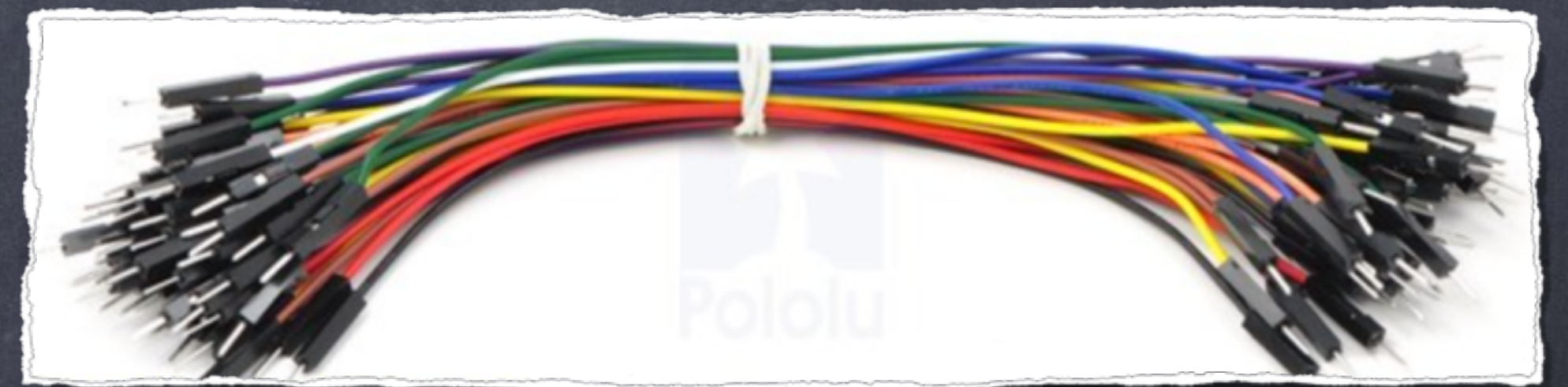
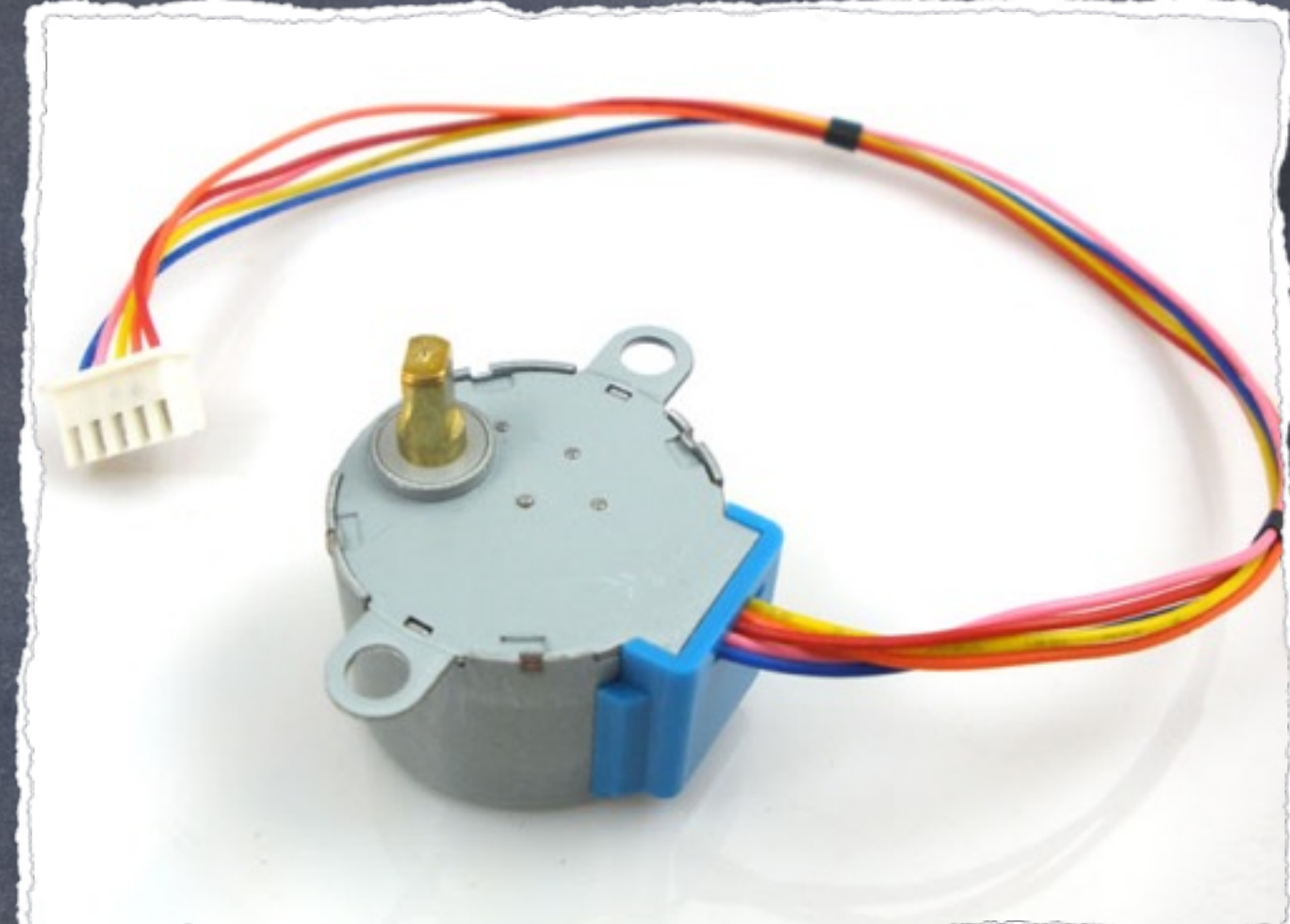
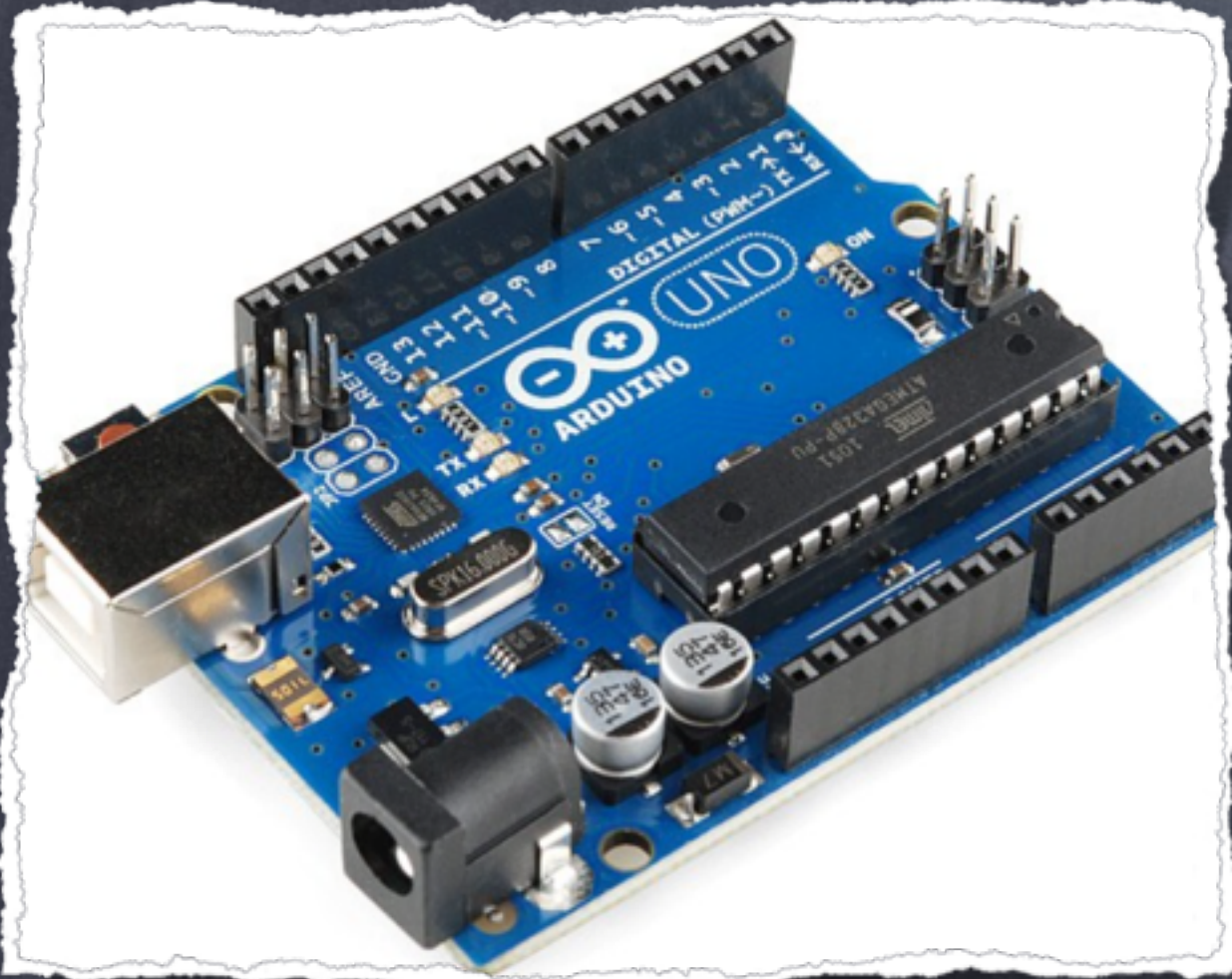
- Ayarlı direnç ile motorun hızını, yönünü ve konumunu kontrol ediniz.

- Açıcılar:

- Sıcaklık algıci ile motorun açısını kontrol ediniz.

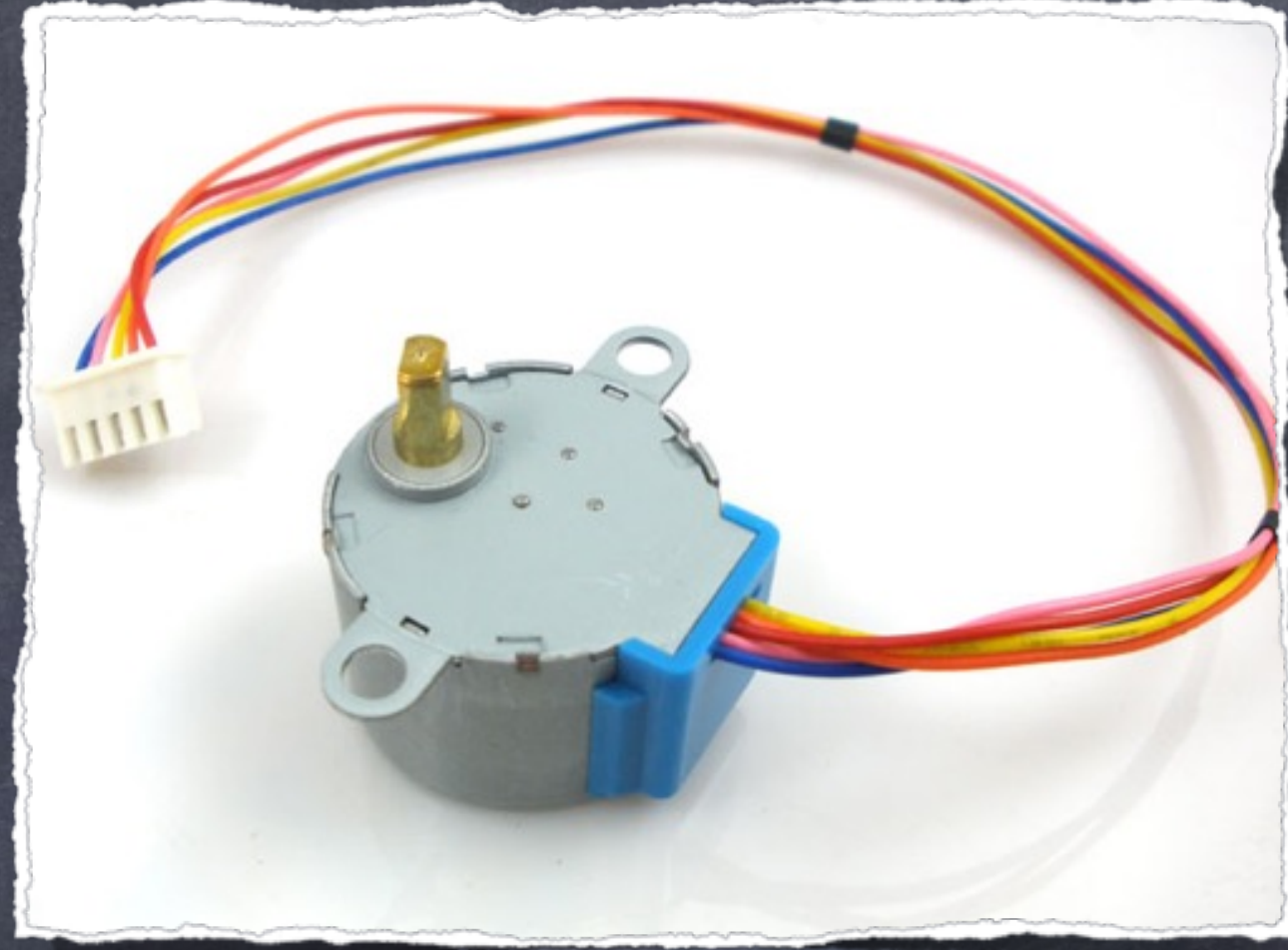


# Adımlıcalar

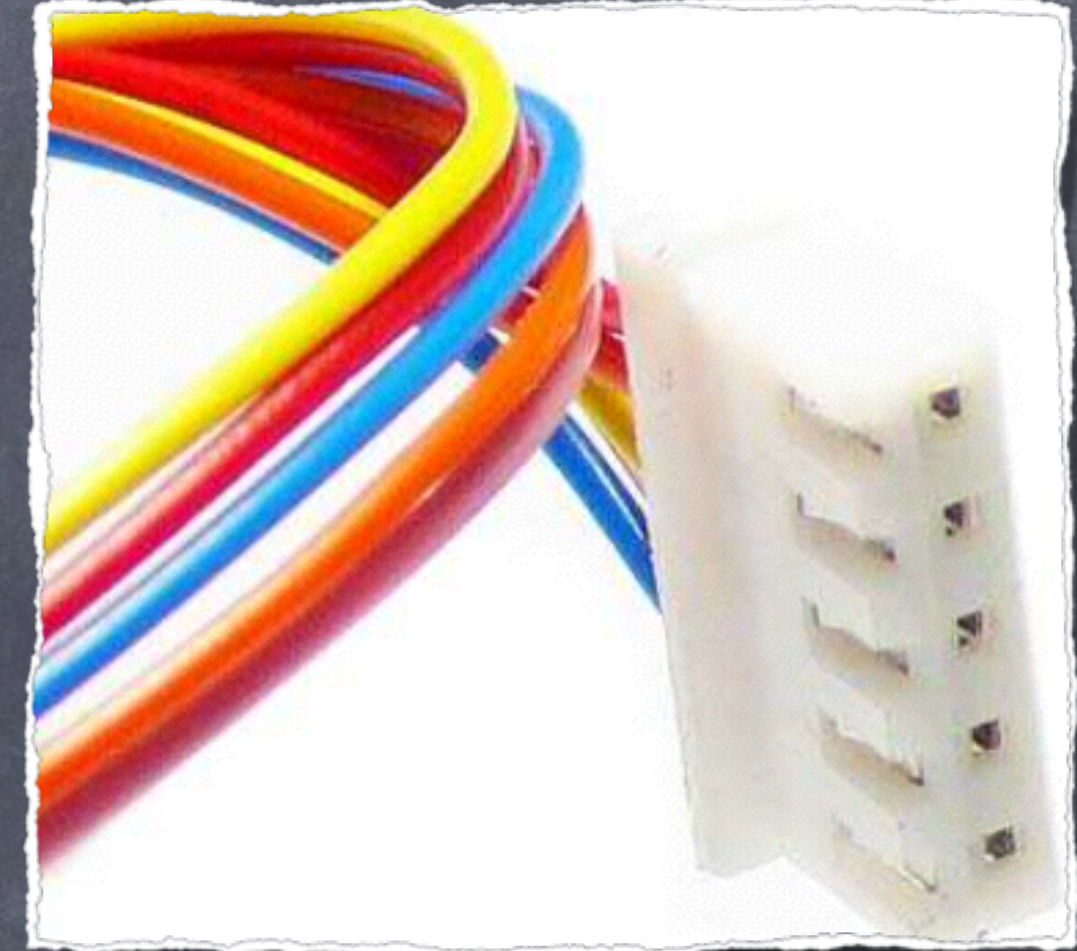




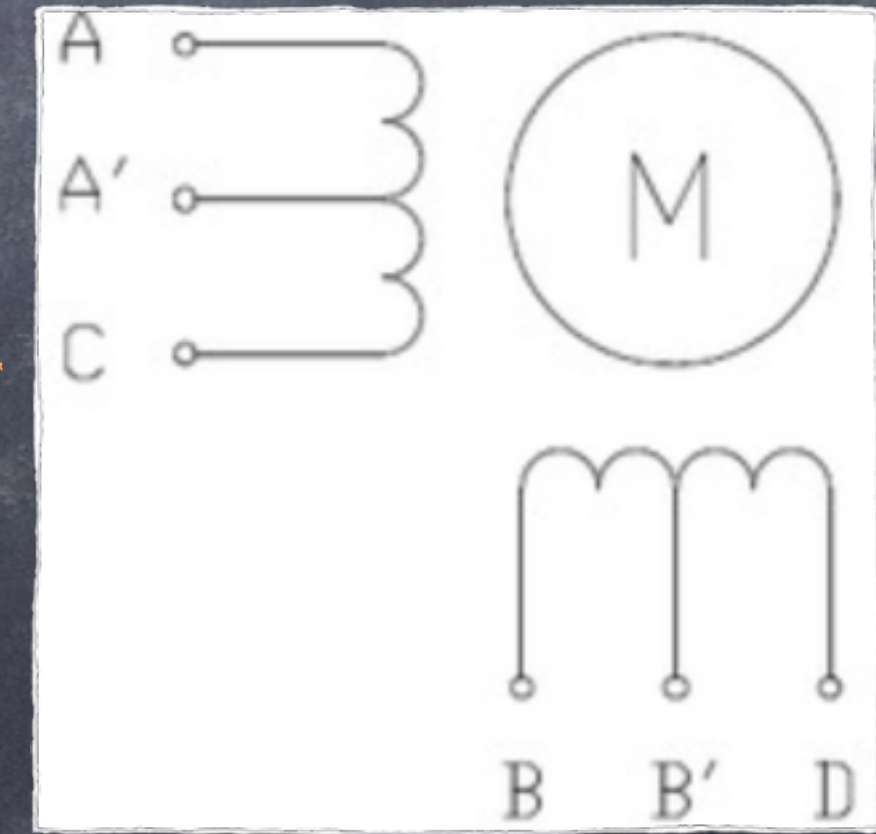
# Adımcılar -Motor-



28BYJ-48 - 5V



Pembe  
Kırmızı  
Turuncu



Sarı  
Kırmızı  
Mavi

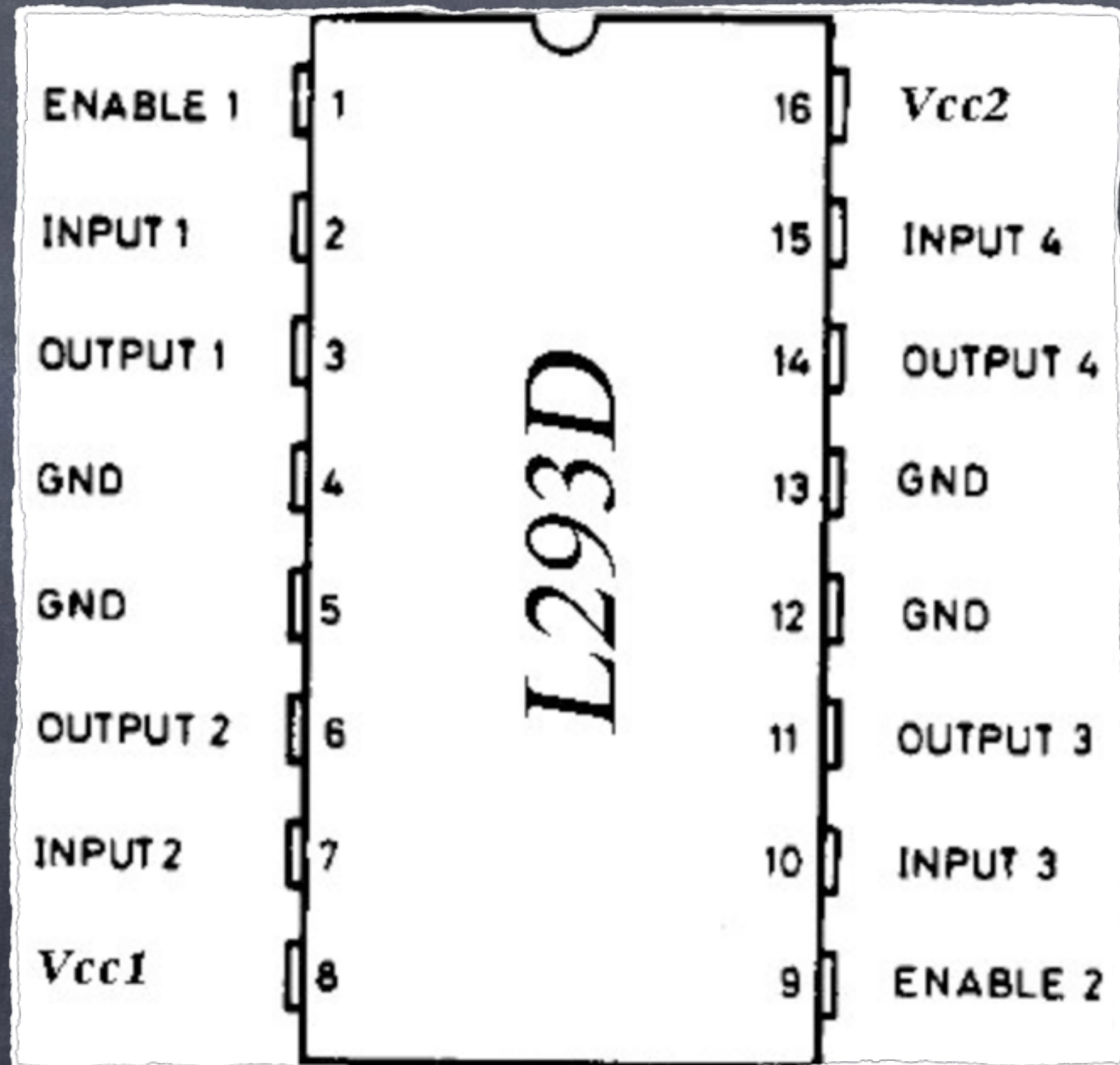
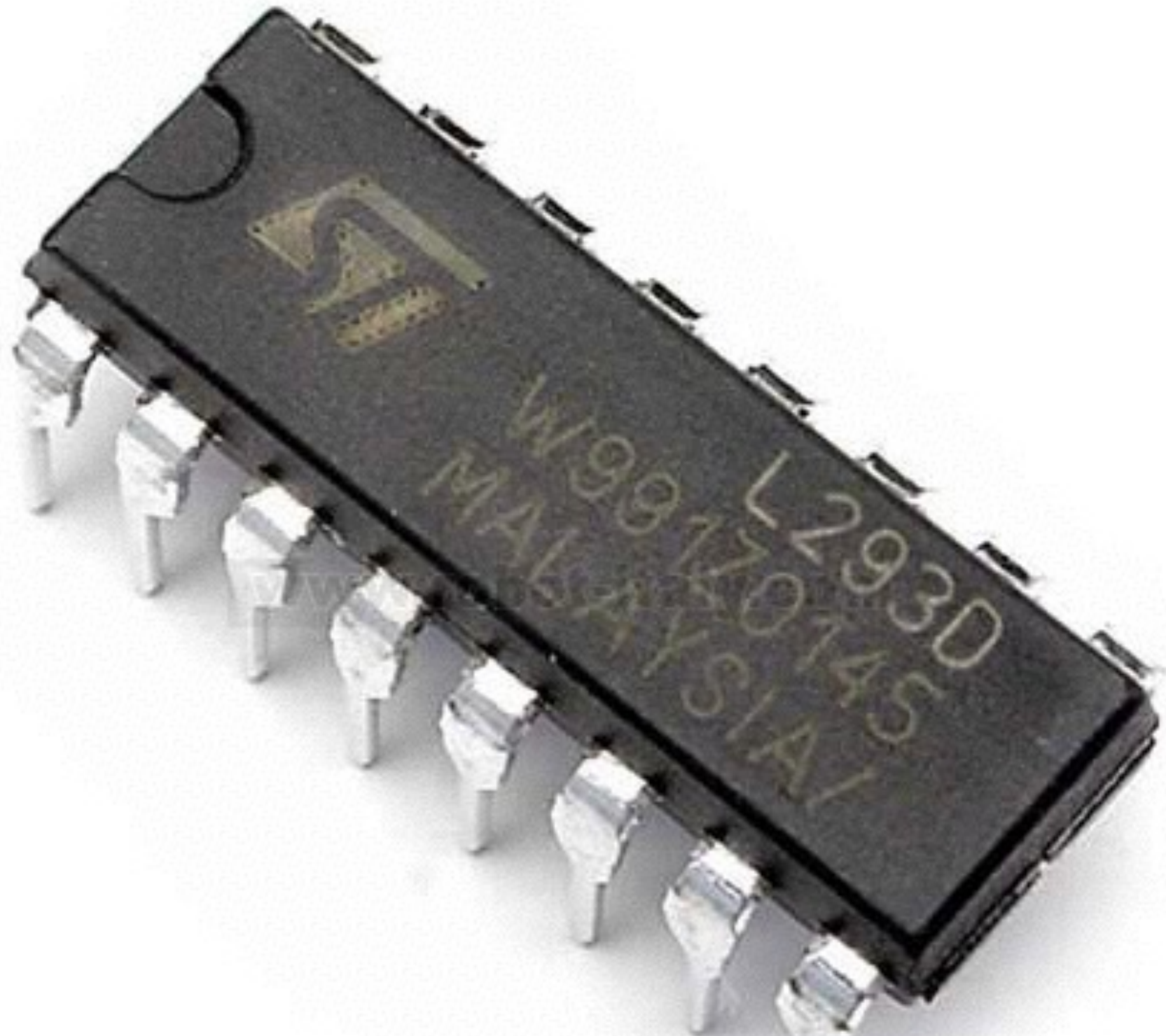
8 adım  
1/64 devir  
düşürücü dişli  
takımı



dönüş başına 512  
adım

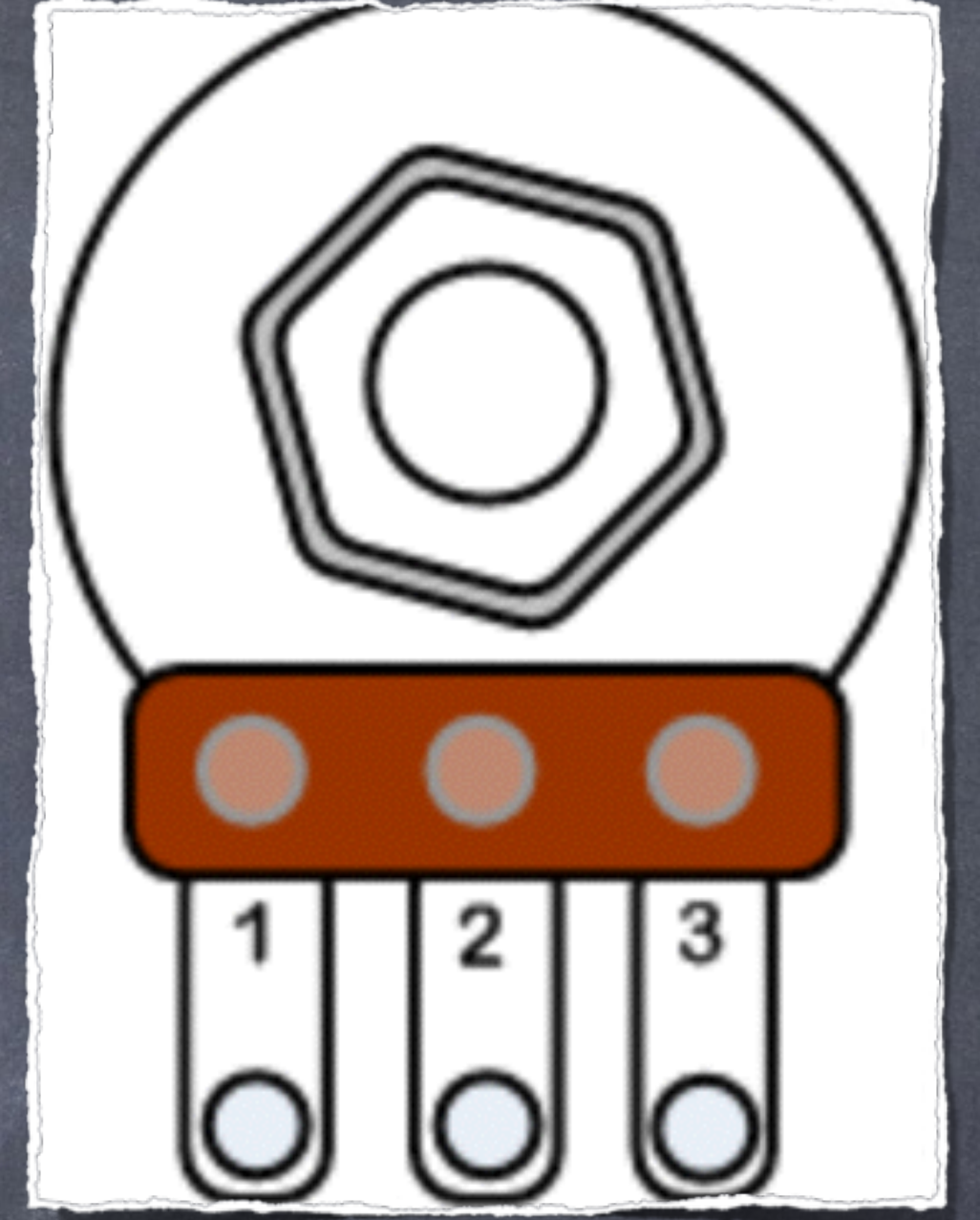


# Adımcılar -Sürücü-





# Adımcılar -Ayarlı direnç-



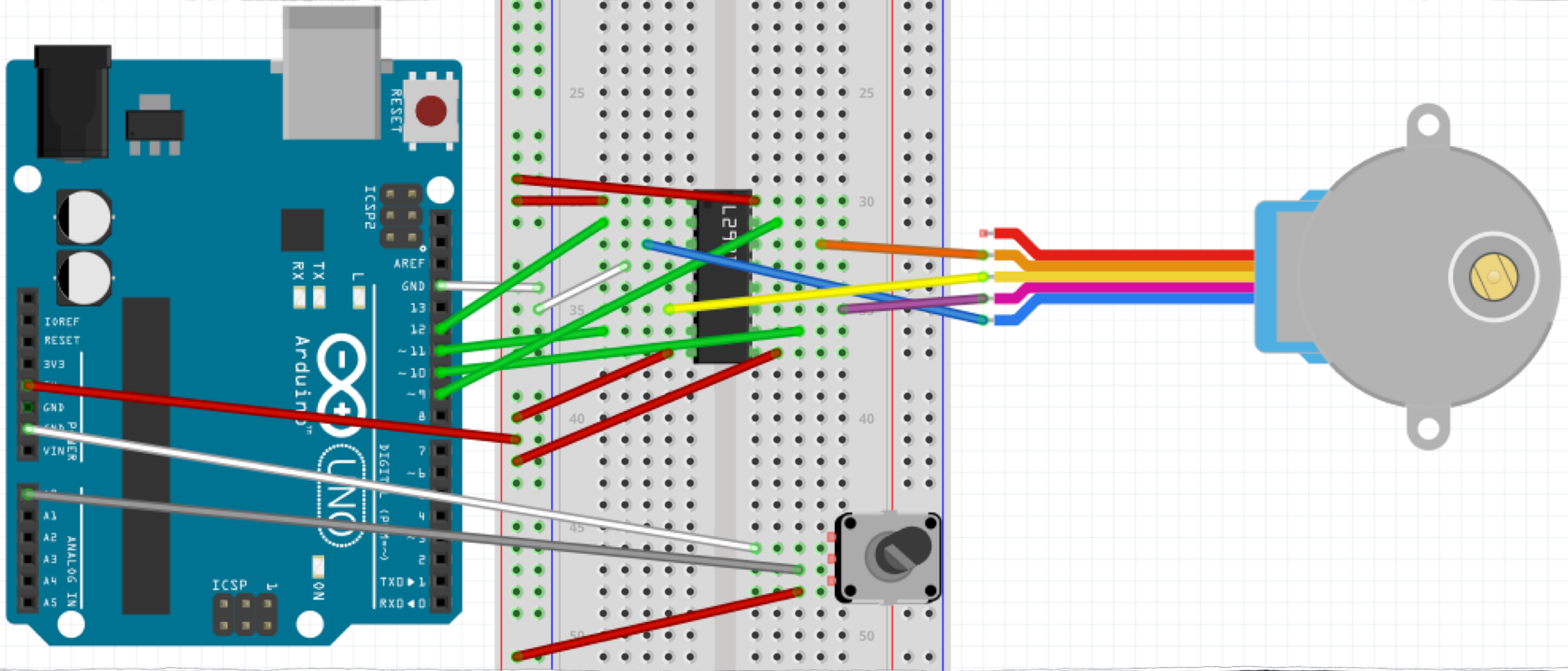
Toprak

Analog Çıkış

5V



# Adımcılar -Bağlantılar-





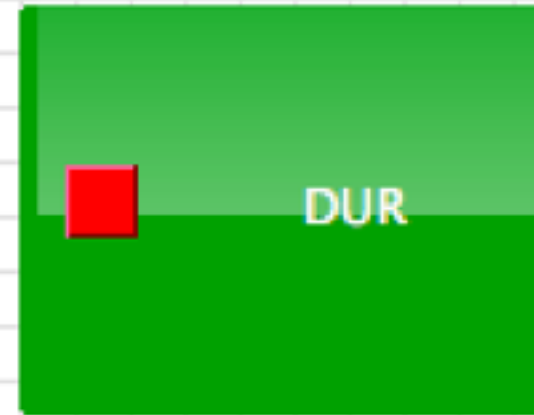
# Adımcılar -Hız yönlendirme-

VISA kaynagi


ASRL1::INSTR

Adım motoru surme bacaklari

9
10
11
12



Hata ciklisi

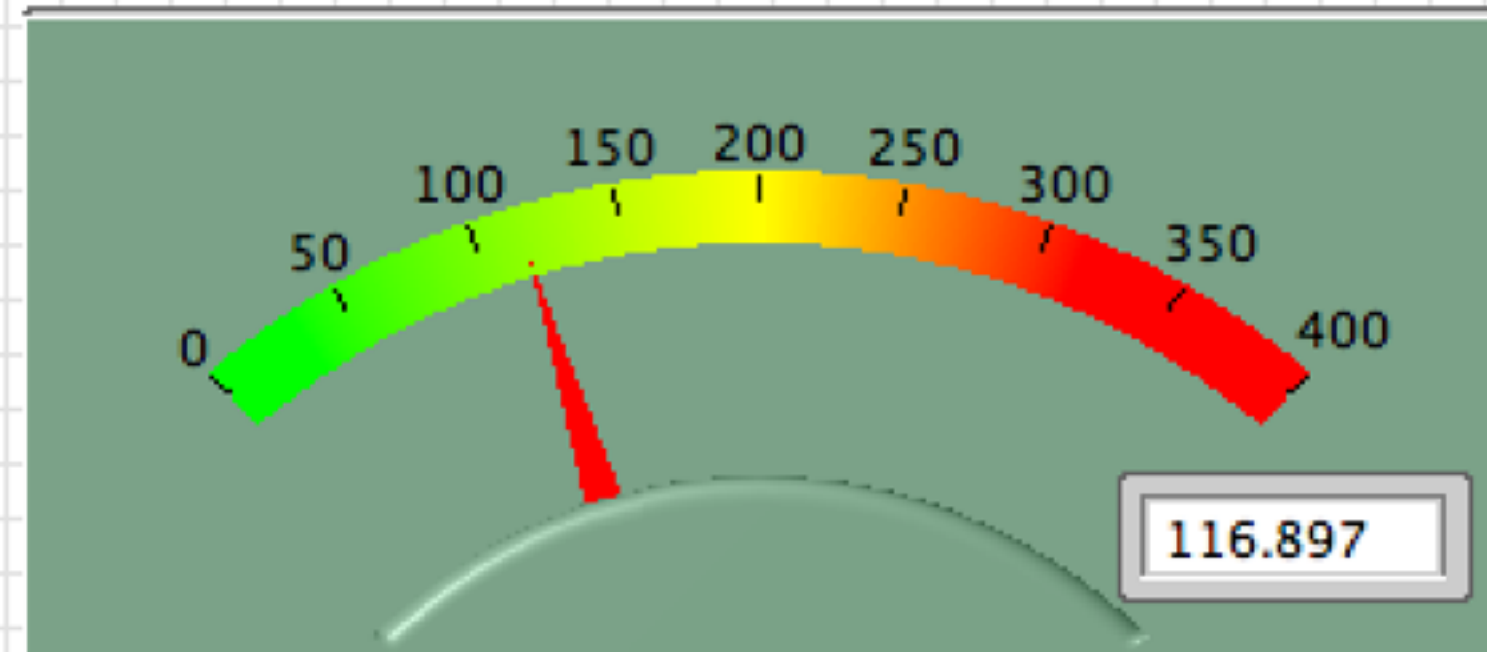
Durum	kod
	d 0
kaynak	

Ayarli direnc okuma bacak numarasi

0

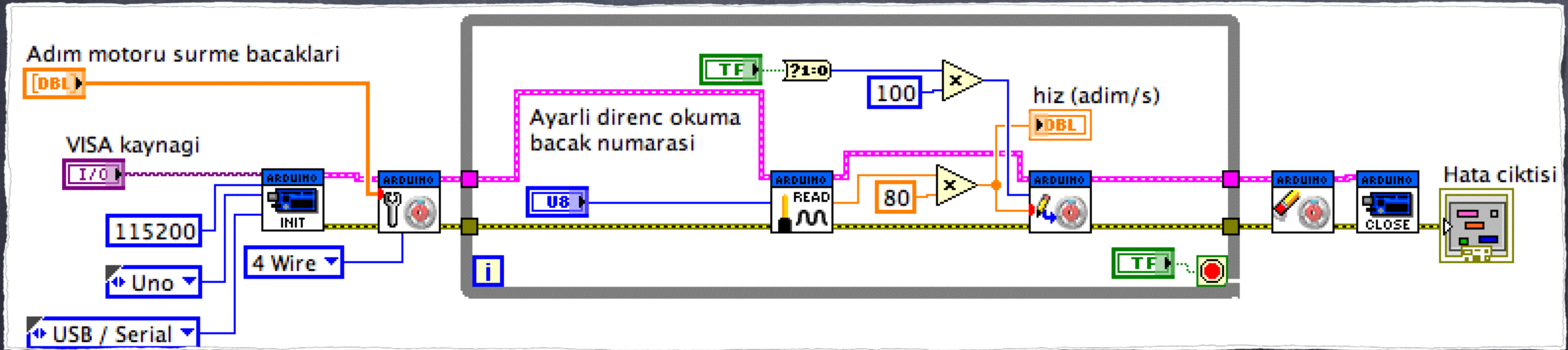


hiz (adim/s)





# Adımcılar -Hız yönlendirme-





# Adımcılar

## -Konum/yön yönlendirme-

VISA kaynagi

ASRL1::INSTR

Adım motoru surme bacaklari

9  
10  
11  
12

Ayarli direnc okuma bacak numarasi

0

Adım motoru donus hizi (adım/s)

350

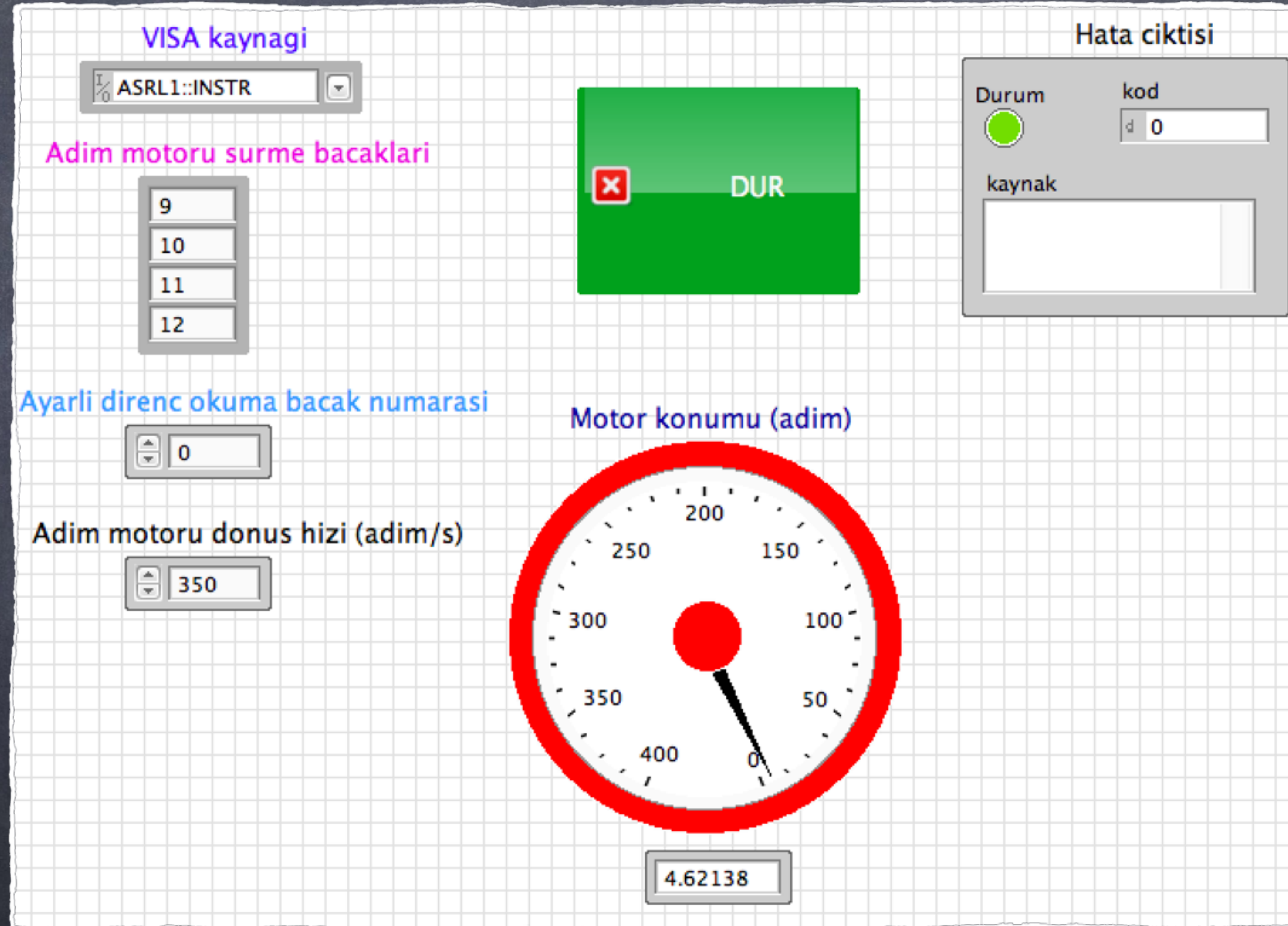
Hata ciklisi

Durum  kod

kaynak

Motor konumu (adım)

4.62138

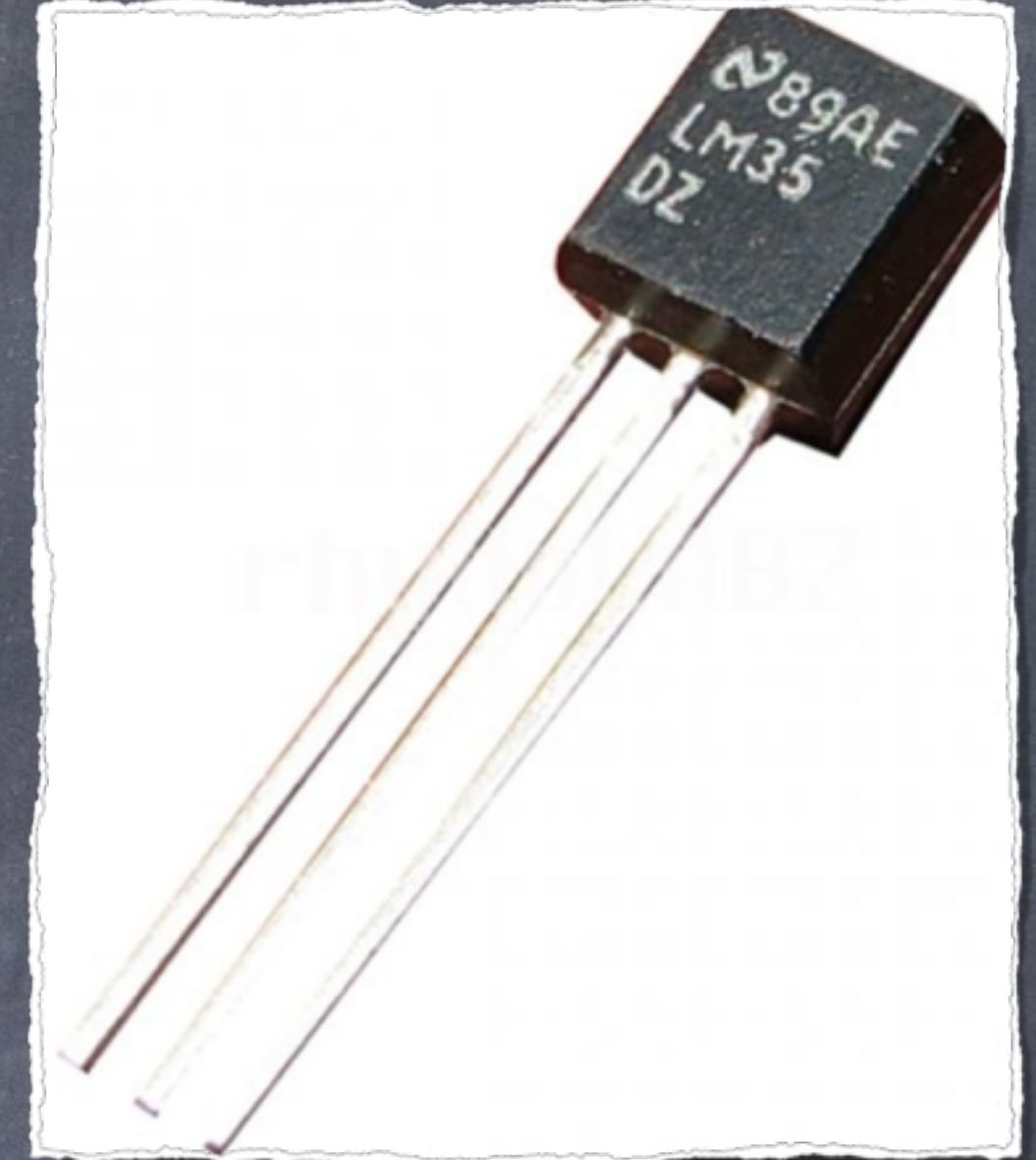
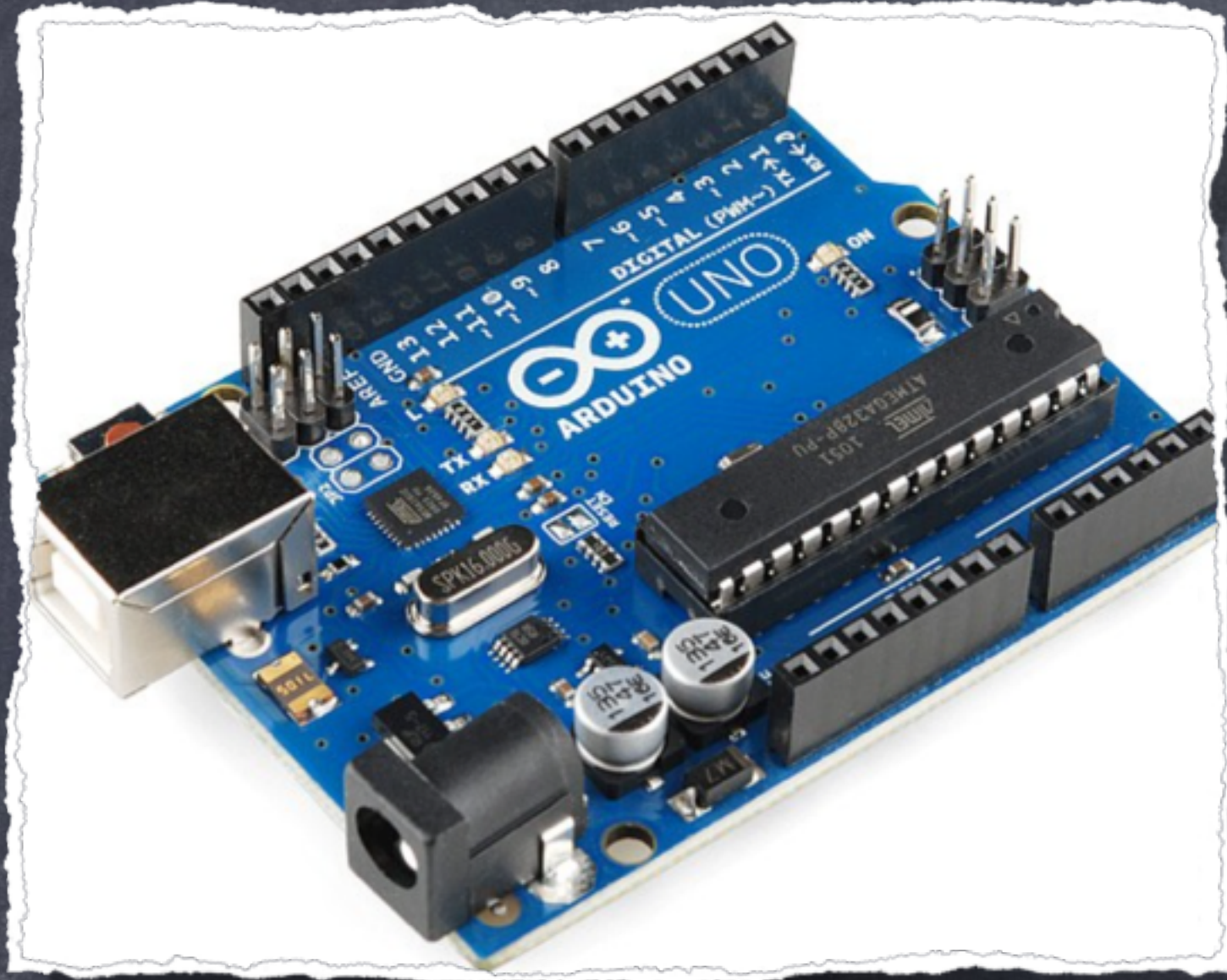






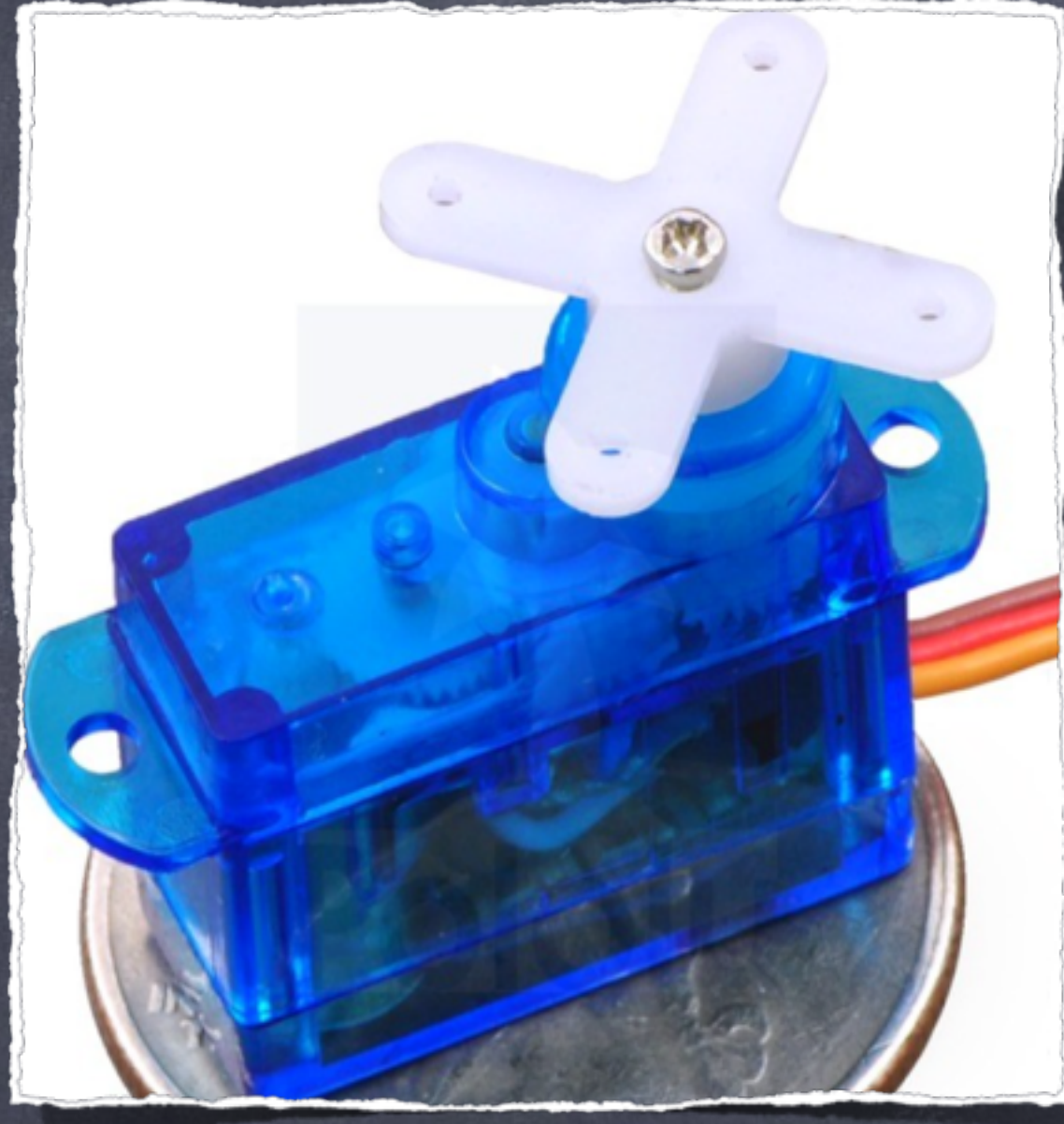


# Ağıcılar





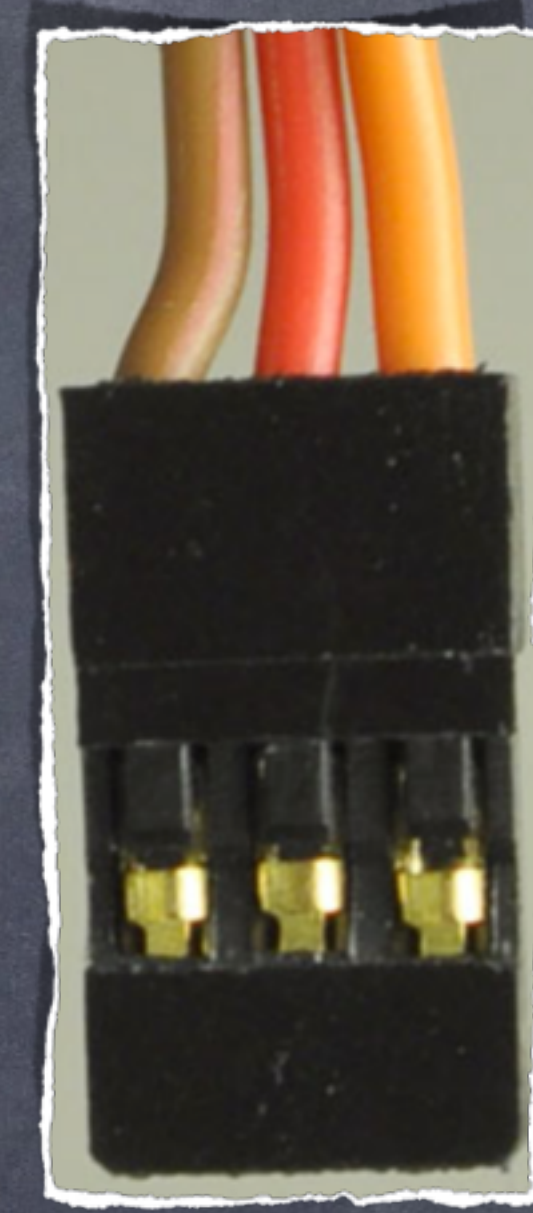
# Açıcılar -Motor-



800-1200 us atma  
genişliği aralığı



0-165 derecelik  
açı aralığı



Toprak

5V

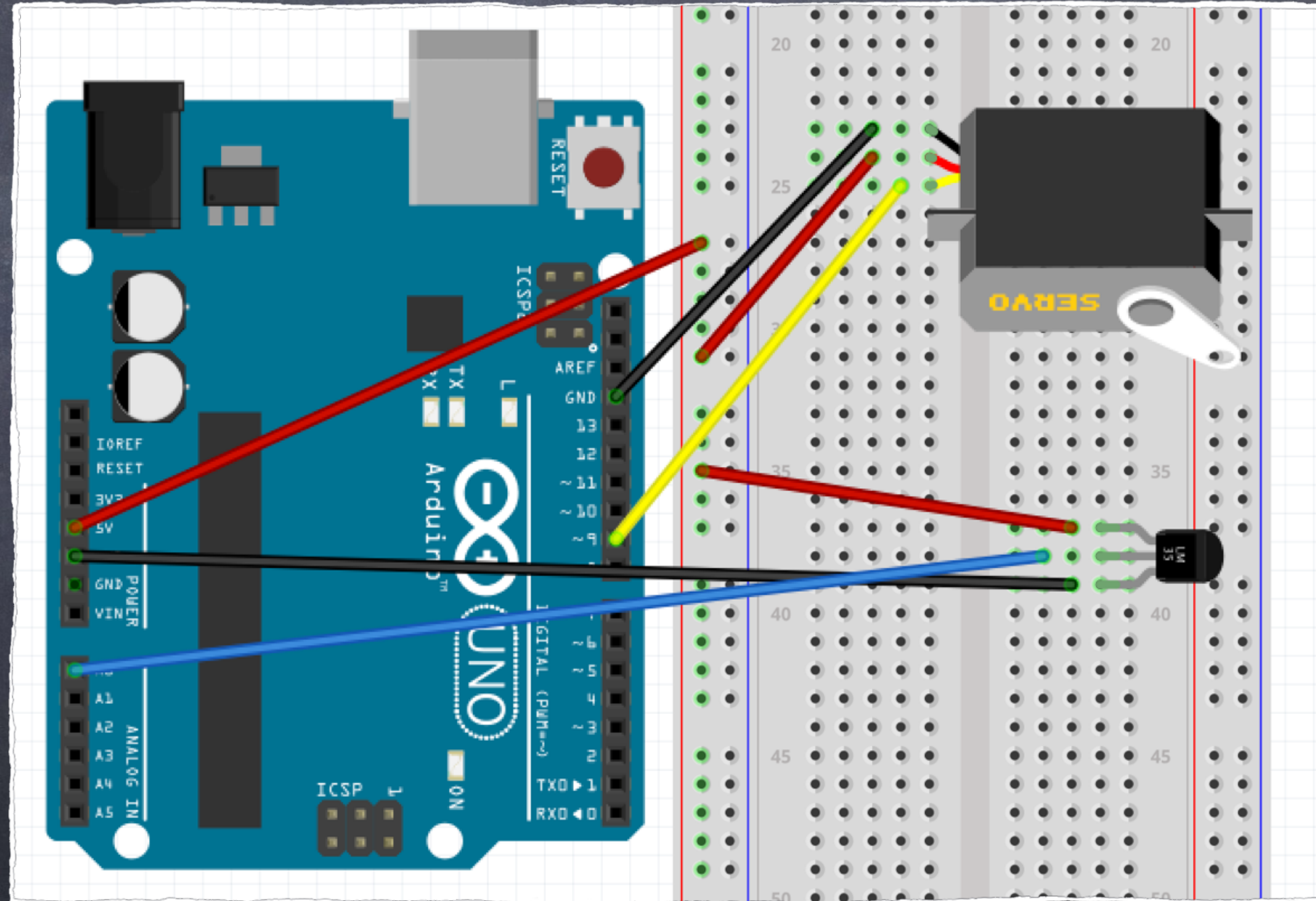
Kontrol







# Açıcılar -Bağlantılar-





# Açıcılar

-Sıcaklık algıci ile açi ayarlama-

VISA kaynagi

ASRL1::INSTR

Acı motoru kontrol bacak numarasi

2

Sıcaklık algıci okuma bacak numarasi

0

Acı degistirme sıcakligi

27


Istenen acı (derece)

Suanki acı (derece)

Sıcaklık (derece)

DUR

Hata ciktili

Durum	kod
	0

kaynak

15

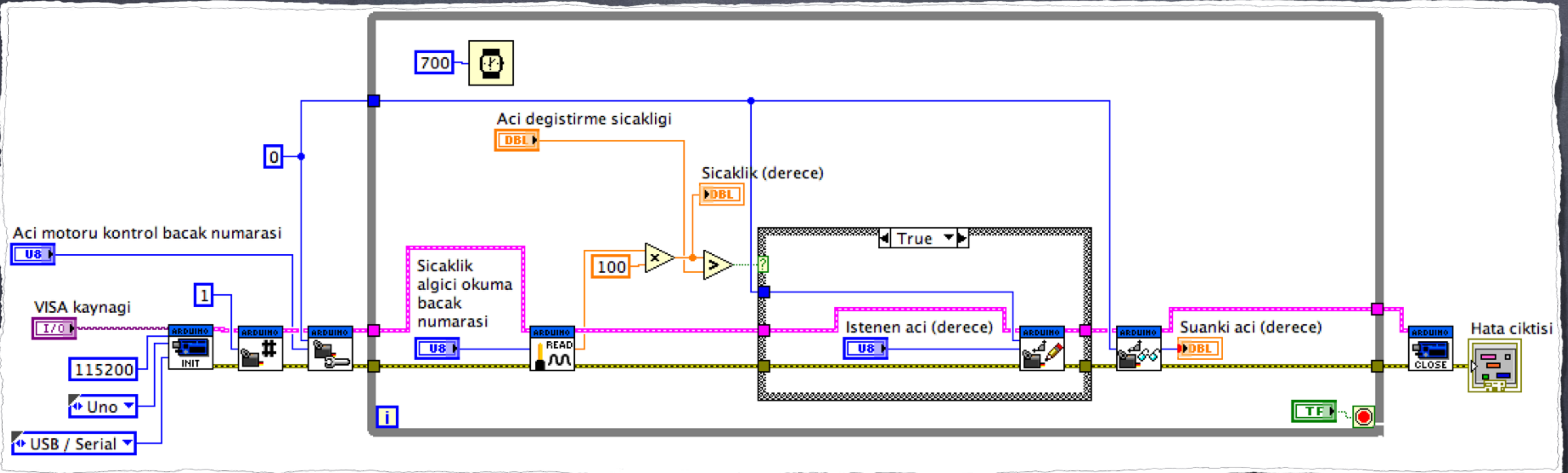
21.483

19.3878



# Açıcılar

-Sıcaklık algıci ile ağı ayarlama-





# Kaynaklar

- University of Oslo, LabVIEW programming I, FYS3240, 2013.
- Hans-Petter Halvorsen, Introduction to LabVIEW, 2012.
- Korolov Igor, How to use LabVIEW, 2011.
- Todd Murphey, LabVIEW for Control Analysis and Design.