İşaret İşleme

Final Öncesi Genel Tekrar3-H15CD3

Dr. Meriç Çetin

versiyon010121

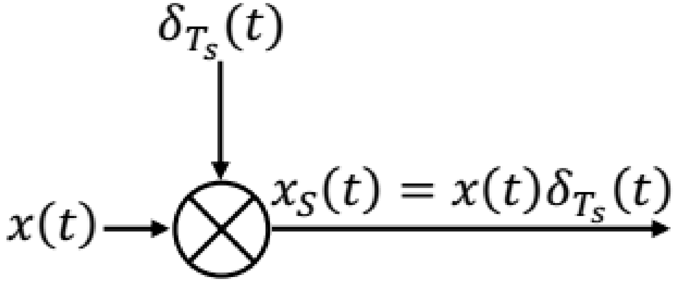
Örnekleme-Sampling

- Örnekleme;
  - sürekli zamanlı sinyalleri işlemek,
  - kaydetmek,
  - iletmek,
  - saklamak ve
  - almak için modern dijital elektroniklerin kullanılmasına izin verir.

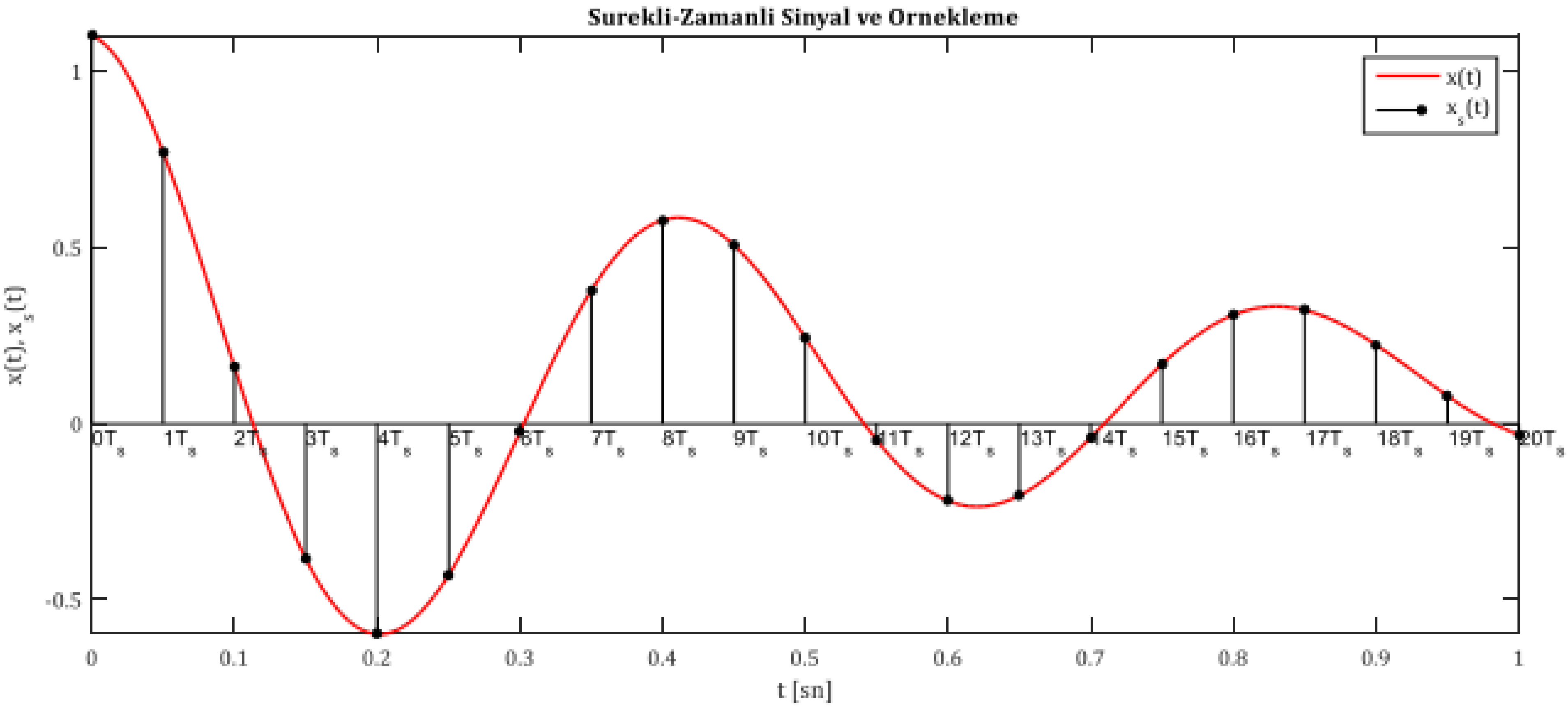
- Neden sinyaller hakkında düşünmek isteyesiniz ki ?

Örnekleme

Örnekleme, dijital sinyal işlemenin temelini oluşturan bir işlem olup zaman domenindeki w_B gibi sonlu bant genişlikli sürekli-zamanlı bir sinyalin T_s periyotlu $\delta_{T_s}(t)$ darbe katarı çarpılarak ayrık-zamanlı hale getirilmesini ve bu sayede dijital sinyal işlemeye uygun hale getirilmesini sağlar. Bunu aşağıdaki şekilde görmek mümkündür.



Aşağıdaki şekilde tipik bir örneklenmiş sinyal görülmektedir.



Örnekleme işlemi sonucunda elde edilen örneklenmiş sinyalin Fourier dönüşümünün bulunmasında, Fourier dönüşümünün

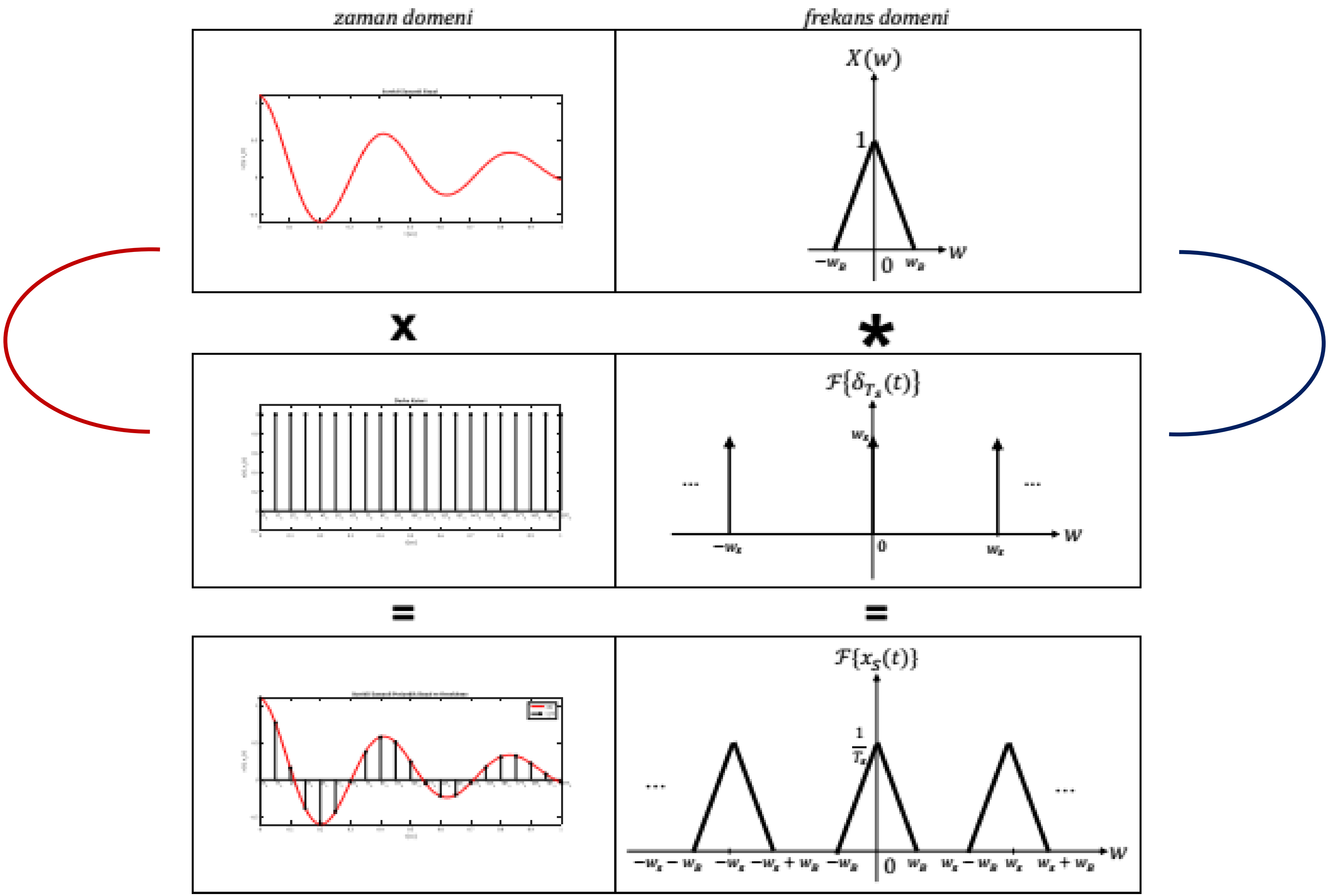
$$x_1(t)x_2(t) \leftrightarrow \frac{1}{2\pi}X_1(w) * X_2(w)$$

şeklindeki çarpma özelliğinden yararlanılır. Örneklenecek $x(t)$ sinyali ile T_s periyotlu $\delta_{T_s}(t)$ darbe katarı sinyalinin çarpılması ile elde edilen örneklenmiş $x_S(t)$ sinyalinin Fourier dönüşümü, çarpma özelliğine göre şu şekilde bulunur:

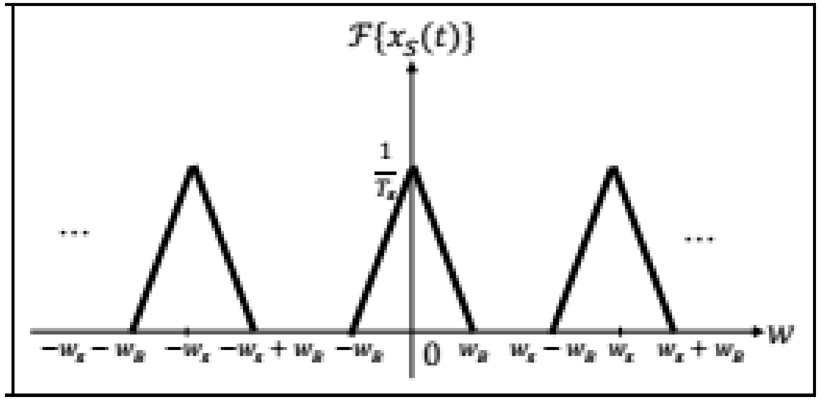
$$x_S(t) = x(t)\delta_{T_s}(t) \leftrightarrow \frac{1}{2\pi}X(w) * \mathcal{F}\{\delta_{T_s}(t)\} = \frac{1}{2\pi}X(w) * w_s \sum_{k=-\infty}^{\infty} \delta(w - kw_s)$$

$$= \frac{1}{T_s}\sum_{k=-\infty}^{\infty} X(w - kw_s)$$

Görüldüğü gibi örneklenmiş sinyalin Fourier dönüşümü, orjinal sinyalin Fourier dönüşümünün tüm frekans domenine yayılmış hali gibidir. Bu durum aşağıdaki şekilde görülmektedir.


zaman domeni
frekans domeni
X(w)
1
−w_B
0
w_B
W
x
*
ℱ{δ_Ts(t)}
w_s
...
−w_s
0
w_s
W
=
=
ℱ{x_S(t)}
1/T_s
...
−w_s − w_B
−w_s
−w_s + w_B
−w_B
0
w_B
w_s − w_B
w_s
w_s + w_B
W

Burada en önemli soru T_s periyodunun nasıl seçileceğidir. Şekle bakıldığında, Fourier dönüşümleri arasında bir girişim ya da örtüşmenin olmaması için

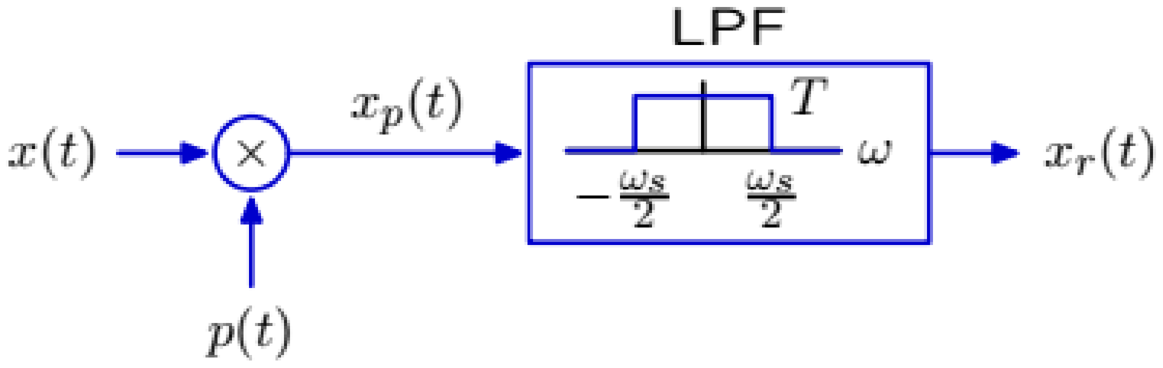


$$w_B < 2w_s$$

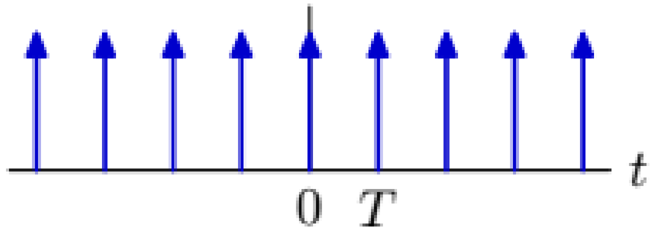
şartının sağlanması gerekir ki bu da örnekleme teoreminin en önemli sonuçlarından biridir. Buna göre, örnekleme frekansı, örneklenecek sinyalin bant genişliğinin en az iki katı olmalıdır.

CT Model of Sampling and Reconstruction

Sampling followed by bandlimited reconstruction is equivalent to multiplying by an impulse train and then low-pass filtering.

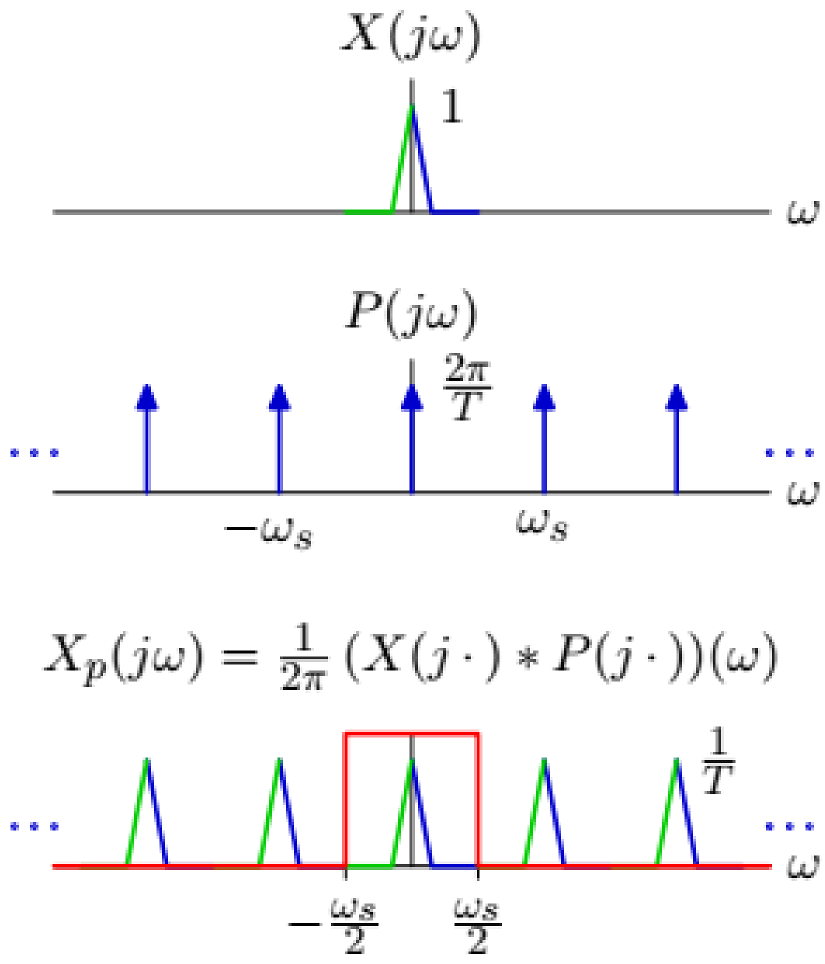


$p(t) =$ "sampling function"



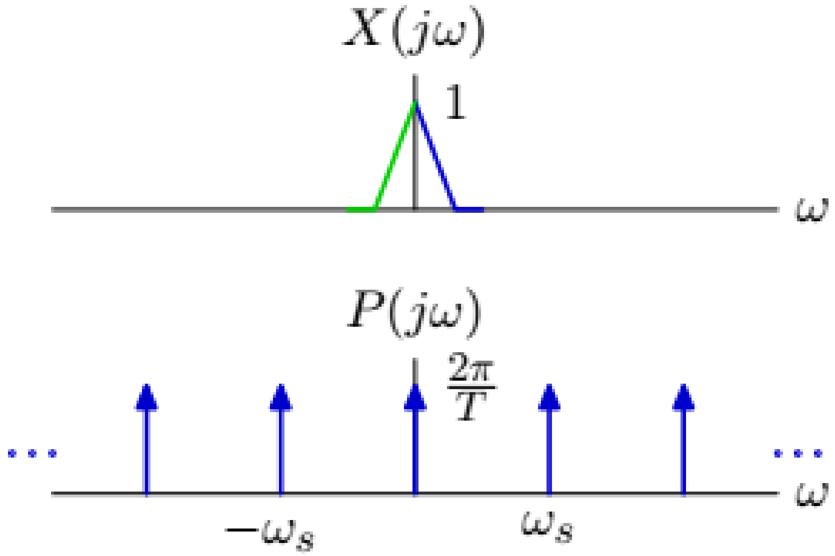
Aliasing

High frequency components of complex signals also wrap.

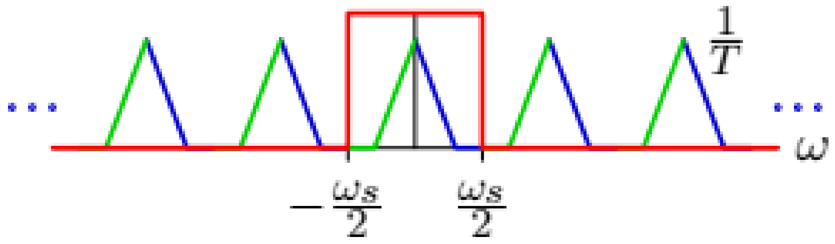


Aliasing

High frequency components of complex signals also wrap.

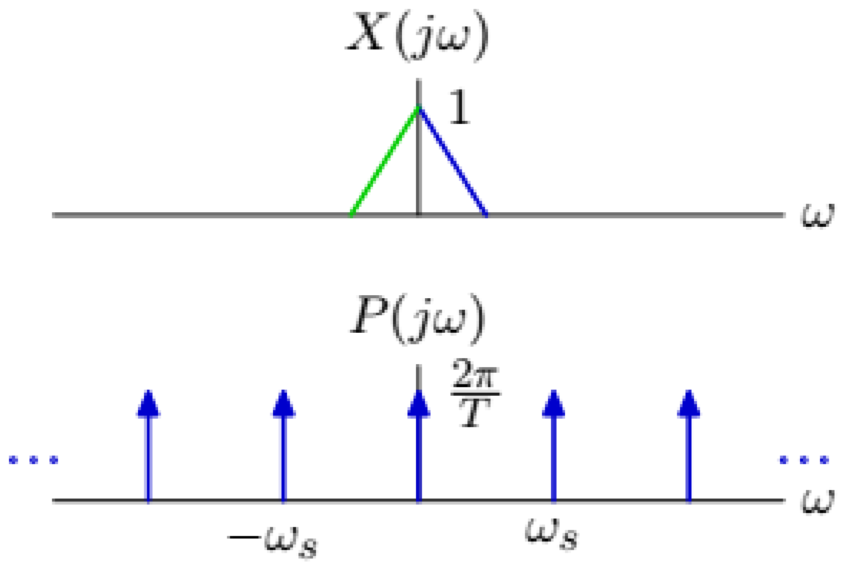


$$X_p(j\omega) = \frac{1}{2\pi}\left(X(j\,\cdot) * P(j\,\cdot)\right)(\omega)$$

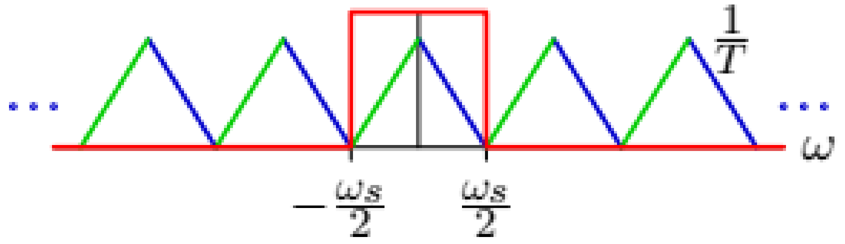


Aliasing

High frequency components of complex signals also wrap.

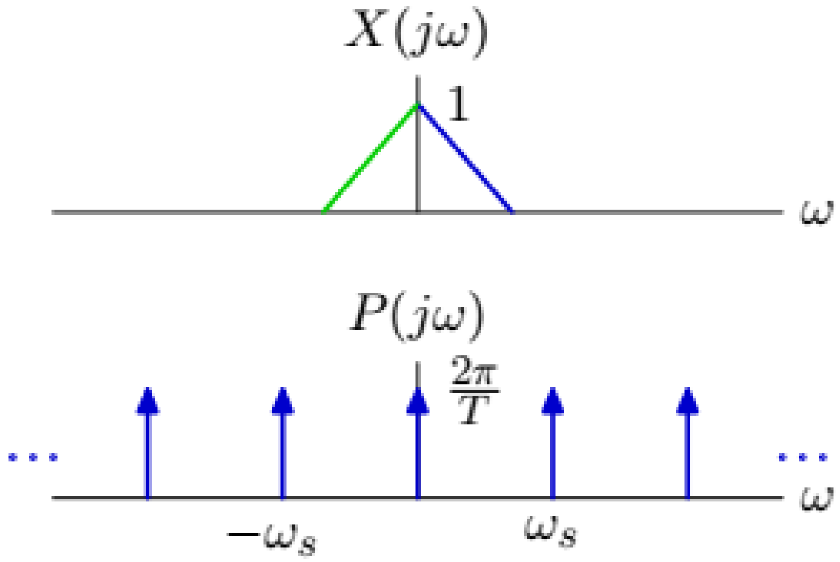


$$X_p(j\omega) = \frac{1}{2\pi}\left(X(j\,\cdot\,) * P(j\,\cdot\,)\right)(\omega)$$

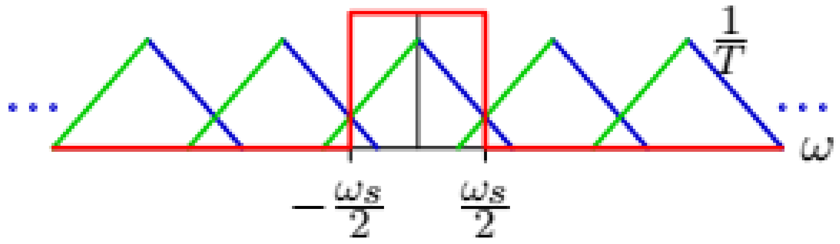


Aliasing

High frequency components of complex signals also wrap.

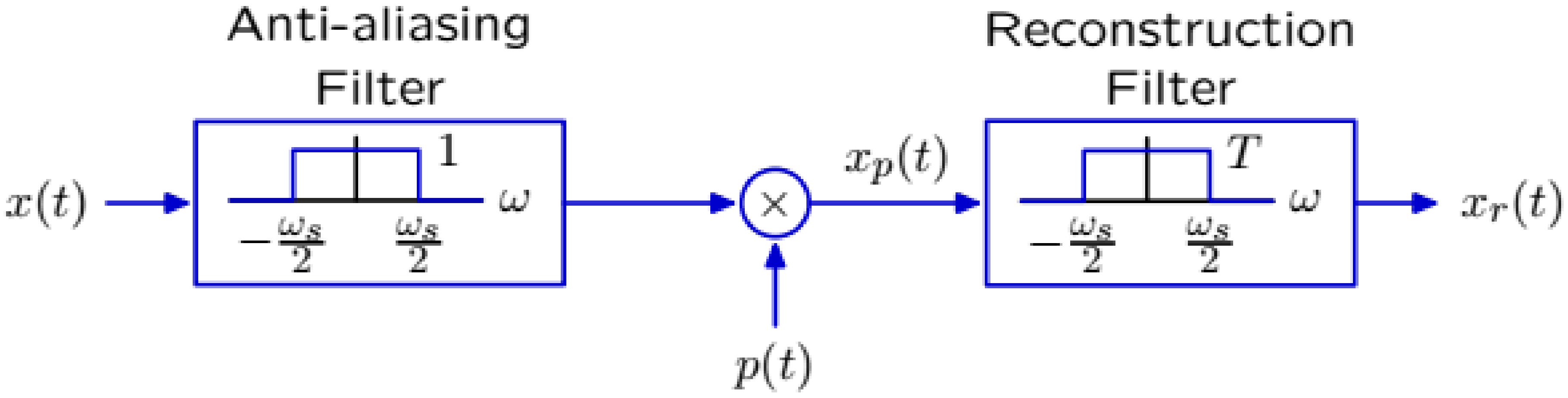


$$X_p(j\omega) = \frac{1}{2\pi}\left(X(j\,\cdot\,) * P(j\,\cdot\,)\right)(\omega)$$



Anti-Aliasing Filter

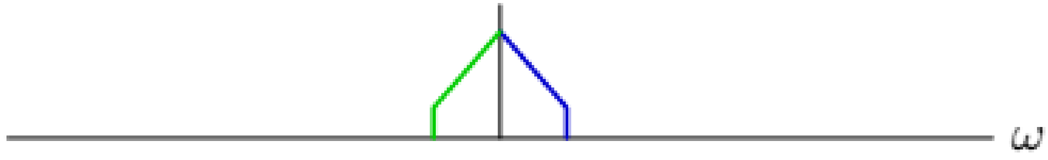
To avoid aliasing, remove frequency components that alias before sampling.



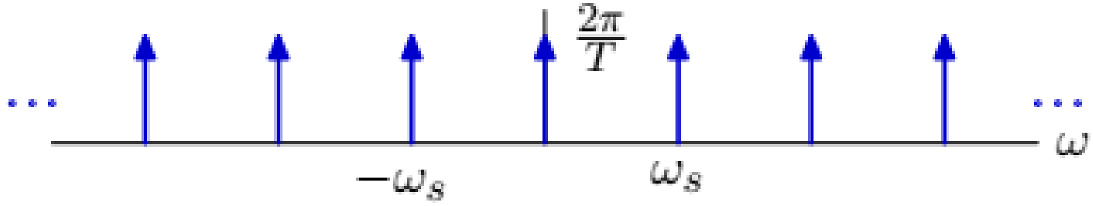
Aliasing

Aliasing increases as the sampling rate decreases.

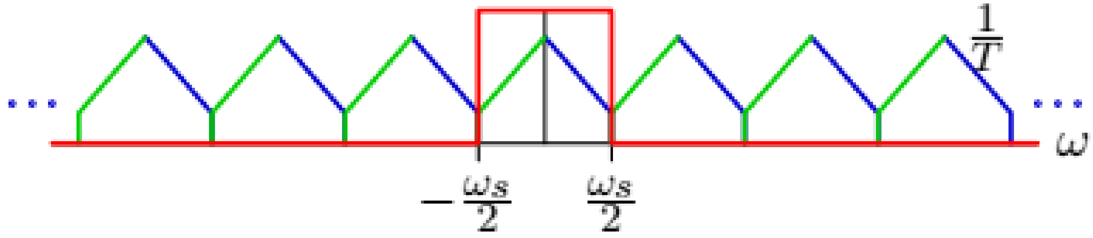
Anti-aliased $X(j\omega)$



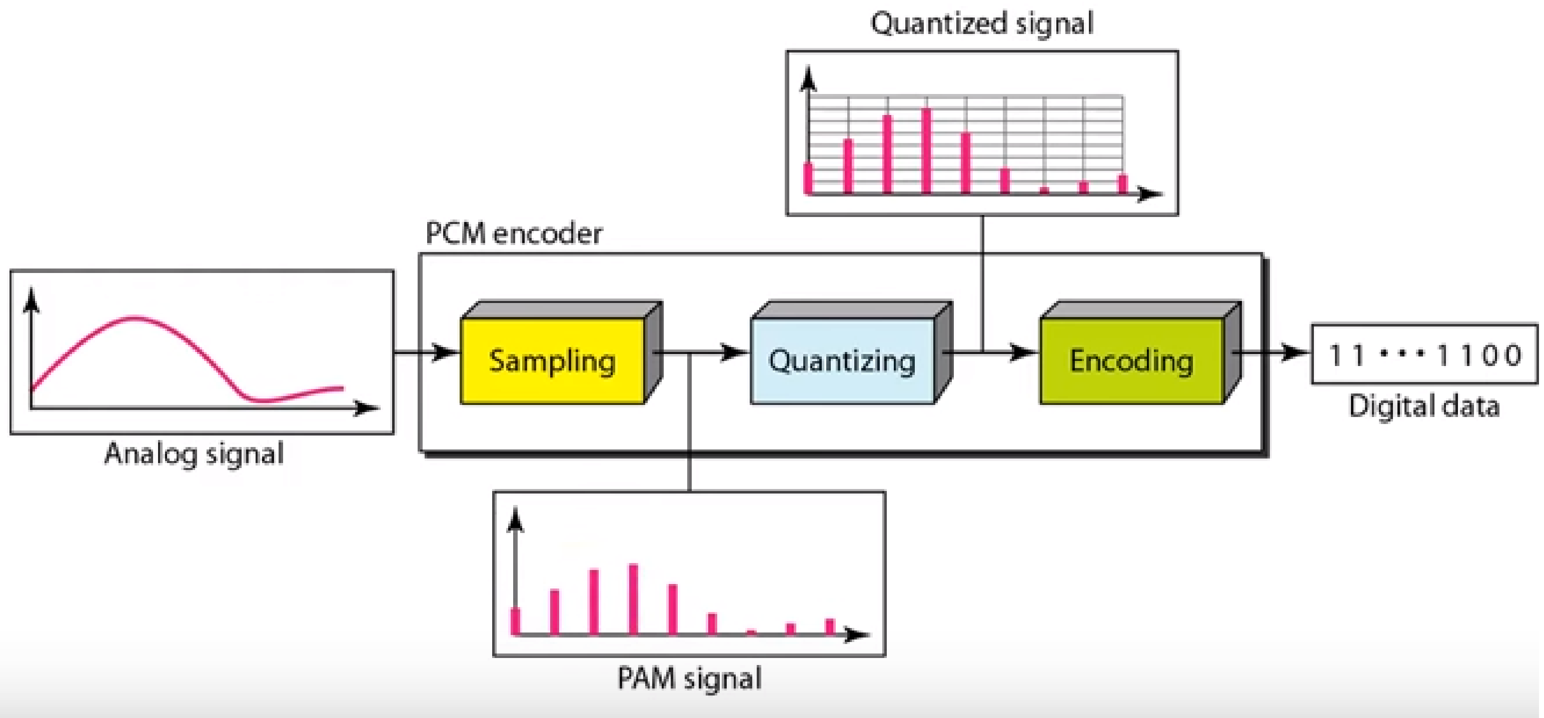
$P(j\omega)$



$$X_p(j\omega) = \frac{1}{2\pi}\left(X(j\,\cdot\,) * P(j\,\cdot\,)\right)(\omega)$$



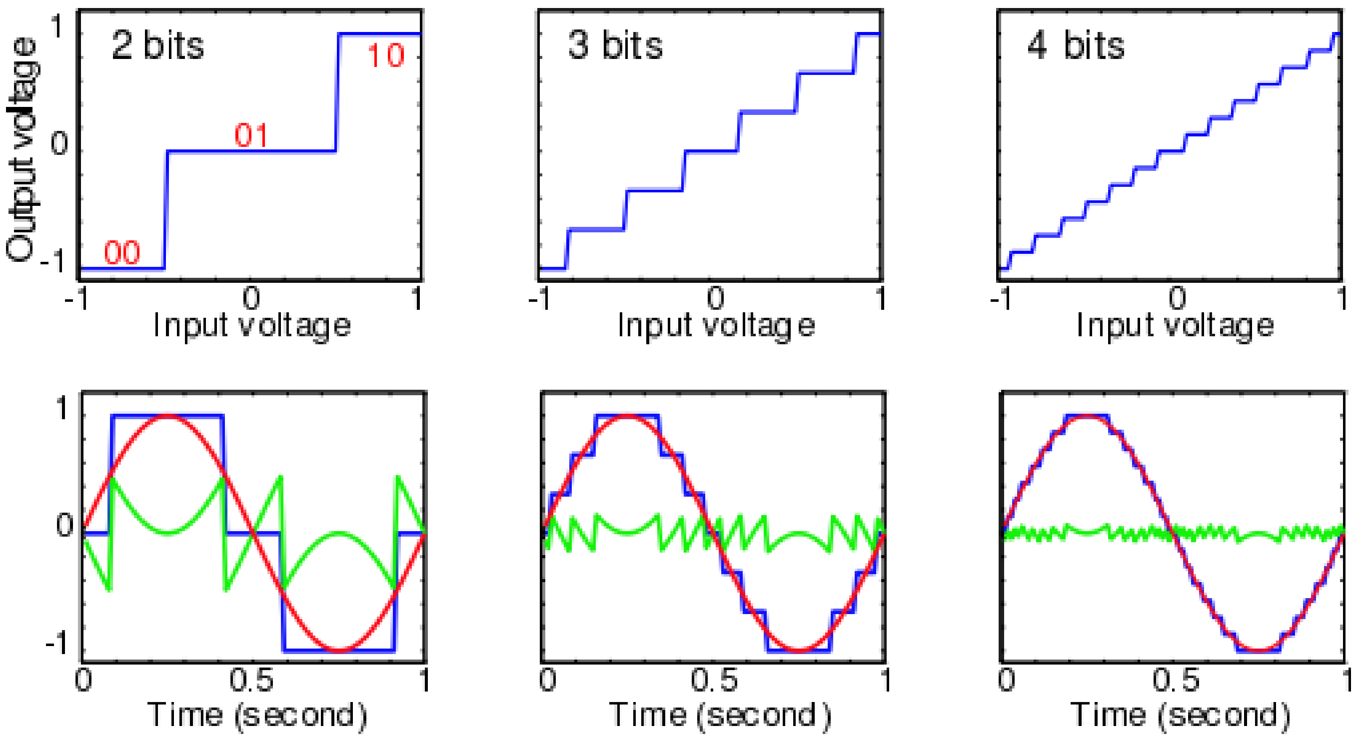
Quantization



- https://www.youtube.com/watch?v=YJmUkNTBa8s

Quantization

We measure discrete amplitudes in bits.



Bit rate = (# bits/sample)×(# samples/sec)

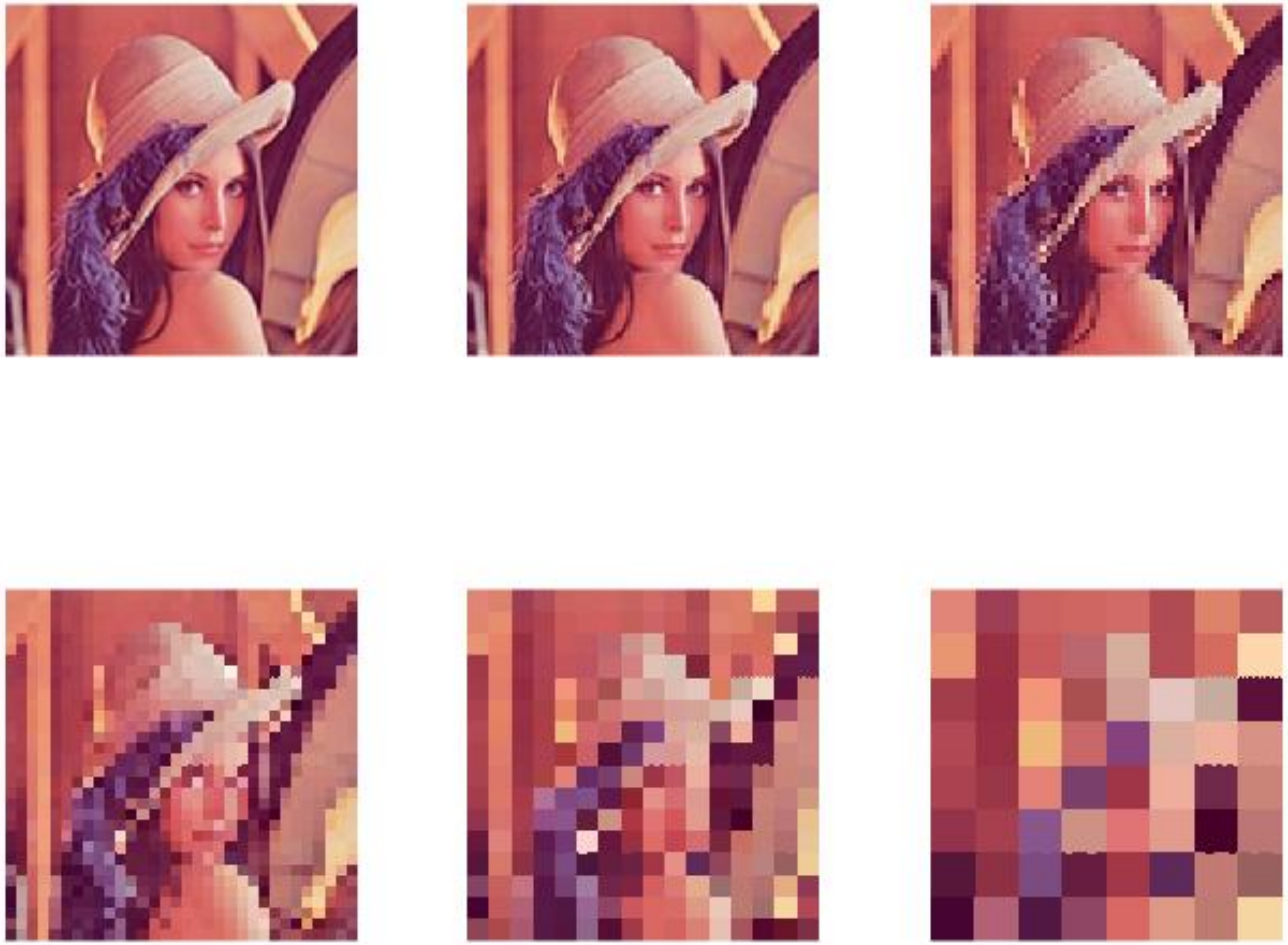
We hear sounds that range in amplitude from 1,000,000 to 1.

How many bits are needed to represent this range?

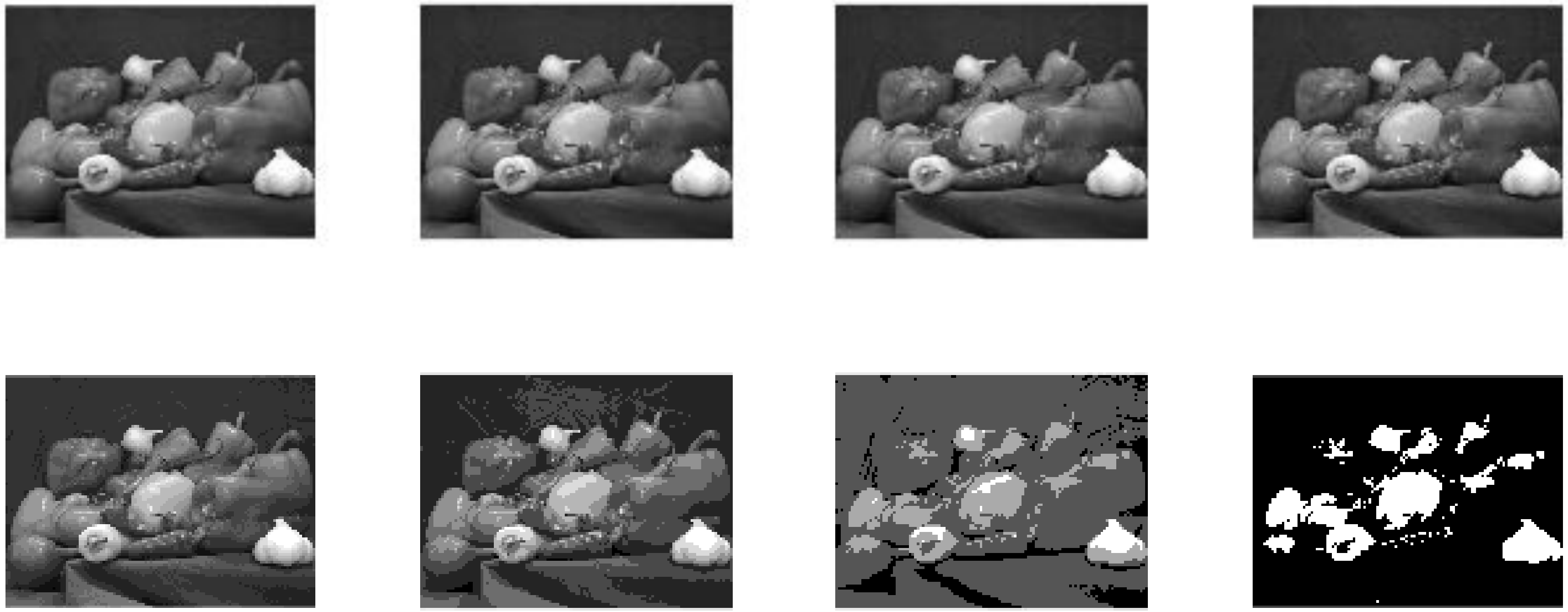
1. 5 bits
2. 10 bits
3. 20 bits
4. 30 bits
5. 40 bits

bits	range
1	2
2	4
3	8
4	16
5	32
6	64
7	128
8	256
9	512
10	1,024
11	2,048
12	4,096
13	8,192
14	16,384
15	32,768
16	65,536
17	131,072
18	262,144
19	524,288
20	1,048,576

2 boyutlu veri için örnekleme



2 boyutlu veri için quantalama



Ayrık Zamanda Durum Uzayı Analizi

Şu ana kadar, doğrusal zamanla-değişmeyen (DZD) sistemlerin giriş-çıkış ilişkileri incelenmiştir. Bu ilişkilerin ortak özelliği bunların *harici tanımlama* olmasıdır yani sadece giriş ve çıkış sinyalleriyle ilgilenilmiş, sistemin içerisinde neler olduğuna bakılmamıştır. Bu bölümde ise, sistemin içerisinde neler olup bittiğini de inceleyen bir *dahili tanımlama* yöntemi olan *durum uzayı* (*state space*) yöntemi ele alınacaktır. Bir DZD sistemin giriş-çıkış ilişkisinin durum uzayı ile gösteriliminin pek çok avantajı vardır; örnek olarak

- Sistemin iç davranışlarını incelemeye imkan sağlar.
- Çok-Girişli Çok-Çıkışlı sistemlerin kolayca incelenmesini sağlar.
- Doğrusal-olmayan ve/veya zamanla-değişen sistemlerin incelenmesine imkan sağlar.
- Bilgisayar ile analiz yapmak çok daha kolaydır.

Durum uzayı analizi matrissel denklemler ve işlemleriyle yapılmaktadır. Bu nedenle, temel lineer cebir konularının bilindiği varsayılacaktır.

Durum Kavramı

Durum: Sürekli-zamanlı (veya ayrık-zamanlı) bir sistemin $t = t_0$ (veya $n = n_0$) anındaki *durumu*, $t \geq t_0$ (veya $n \geq n_0$) anlarındaki giriş sinyalinin bilinmesi koşuluyla, sistemin $t \geq t_0$ (veya $n \geq n_0$) anındaki durumunu ve çıkışını belirlemede yeterli olan minimum bilgi olarak tanımlanır.

Durum Değişkeni: Bu bilgiyi taşıyan değişkenlere *durum değişkenleri* denir ve bu derste q_1, q_2,..., q_N gibi değişkenlerle gösterilecektir. Not edilmelidir ki bu tanım sadece nedensel sistemler için geçerlidir. Durum değişkenleri fiziksel olarak ölçülebilen ya da gözlenebilen büyüklükler olmak zorunda değildir. Bu nedenle durum değişkenlerinin seçiminin serbest olması durum uzayı analizinde büyük avantaj sağlar. Yine de, durum değişkenlerini ölçülebilir büyüklüklerden seçmek pratik olarak kolaylık sağlar.

Durum Kavramı-devam

Durum Vektörü: Eğer bir sistemin durumunu göstermek için N adet durum değişkeni gerekiyor ise o zaman bu N adet durum değişkeni $\mathbf{q}$ gibi bir vektörün N adet elemanı olarak görülebilir, yani

$$\mathbf{q} = \begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_N \end{bmatrix}$$

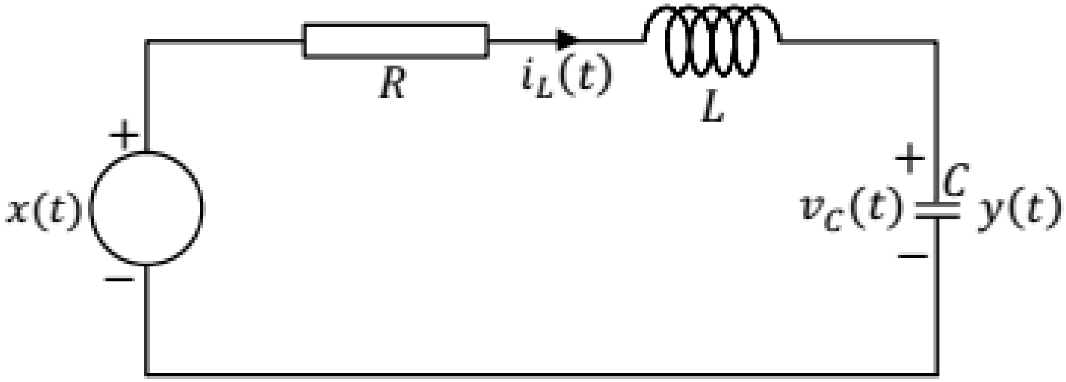
şeklinde gösterilebilir. Bu şekildeki vektöre *durum vektörü* denir.

Durum Uzayı: Eksenleri q_1, q_2,..., q_N gibi durum değişkenlerinden oluşan N-boyutlu uzaya *durum uzayı* denir.

Durum Kavramı-devam

Durum-Uzayı Denklemleri: Durum uzayı analizinde, giriş değişkeni, durum değişkenleri ve çıkış değişkeni olmak üzere üç tip değişken vardır. Bu derste, x giriş sinyalini, q durum değişkenini ve y de çıkış sinyalini temsil edecektir. Sürekli-zamanlı ve ayrık-zamanlı sistemler için ayrı ayrı incelenecektir. Bir sisteme ilişkin dinamik denklemler yazıldıktan sonra uygun durum değişkenleri seçilerek elde edilen denklemlere *durum denklemleri*, sistemin durumlarını kullanarak çıkış sinyalini veren denklemede *çıkış denklemi* adı verilmektedir. Durum denklemleri ve çıkış denkleminden oluşan denklem kümesine *durum uzayı denklemleri* adı verilmektedir

Örnek-devam



Bu denklemler düzenlenirse,

$$\dot{q}_1(t) = -\frac{R}{L}q_1(t) - \frac{1}{L}q_2(t) + \frac{1}{L}x(t)$$
$$\dot{q}_2(t) = \frac{1}{C}q_1(t)$$
$$y(t) = q_2(t)$$

elde edilir. Bu denklemler matrissel olarak

durum denklemleri

$$\begin{bmatrix}\dot{q}_1(t)\\ \dot{q}_2(t)\end{bmatrix} = \begin{bmatrix}-\frac{R}{L} & -\frac{1}{L}\\ \frac{1}{C} & 0\end{bmatrix}\begin{bmatrix}q_1(t)\\ q_2(t)\end{bmatrix} + \begin{bmatrix}\frac{1}{L}\\ 0\end{bmatrix}x(t)$$

çıkış denklemi

$$y(t) = \begin{bmatrix}0 & 1\end{bmatrix}\begin{bmatrix}q_1(t)\\ q_2(t)\end{bmatrix}$$

gibi ifade edilebilir.

Ayrık Zamanlı Sistemlerde Durum Uzayı Denklemleri ve Çözümleri

Ayrık Zamanlı Sistemlerde Durum Uzayı Denklemleri

Eğer bir ayrık-zamanlı sistemin durumunu göstermek için N adet durum değişkeni gerekiyor ise o zaman bu sistemin durum uzayı denklemleri, durum denklemleri ve çıkış denklemi olmak üzere iki kısımdan oluşur. Durum uzayı denklemlerinde, durum denklemleri,

$$
\begin{gathered}
q_1[n+1] = a_{11}q_1[n] + a_{12}q_2[n] + \cdots + a_{1N}q_N[n] + b_1x[n] \\
q_2[n+1] = a_{21}q_1[n] + a_{22}q_2[n] + \cdots + a_{2N}q_N[n] + b_2x[n] \\
\vdots \\
q_n[n+1] = a_{N1}q_1[n] + a_{N2}q_2[n] + \cdots + a_{NN}q_N[n] + b_nx[n]
\end{gathered}
$$

şeklinde veya $\mathbf{q}[n] = \begin{bmatrix} q_1[n] \\ q_2[n] \\ \vdots \\ q_N[n] \end{bmatrix}$, $\mathbf{A} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{bmatrix}$ ve $\mathbf{B} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_N \end{bmatrix}$ olmak üzere matrissel olarak

$$
\mathbf{q}[n+1] = \mathbf{A}\mathbf{q}[n] + \mathbf{B}x[n]
$$

Ayrık Zamanlı Sistemlerde Durum Uzayı Denklemleri

çıkış denklemi,

$$y[n] = c_1 q_1[n] + c_2 q_2[n] + \cdots + c_N q_N(t) + \mathbf{D}x[n]$$

şeklinde veya $\mathbf{C} = [c_1 \quad c_2 \quad \cdots \quad c_n]$ ve $\mathbf{D} = d$ olmak üzere matrissel olarak

$$y[n] = \mathbf{Cq}[n] + \mathbf{D}x[n]$$

biçiminde yazılabilir. Dolayısıyla ayrık-zamanlı DZD bir sistemim durum uzayı denklemleri $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ matris kümesi ile

$$\mathbf{q}[n+1] = \mathbf{Aq}[n] + \mathbf{B}x[n]$$
$$y[n] = \mathbf{Cq}[n] + \mathbf{D}x[n]$$

şeklinde ifade edilebilir. $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ matris kümesi bu sistemin durum uzayı gösterilimidir. Bir sistemin durum uzayı gösterilimi tek değildir yani aynı sisteme ait çok sayıda farklı durum uzayı gösterilimi yani $\{\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}\}$ matris kümesi olabilir.

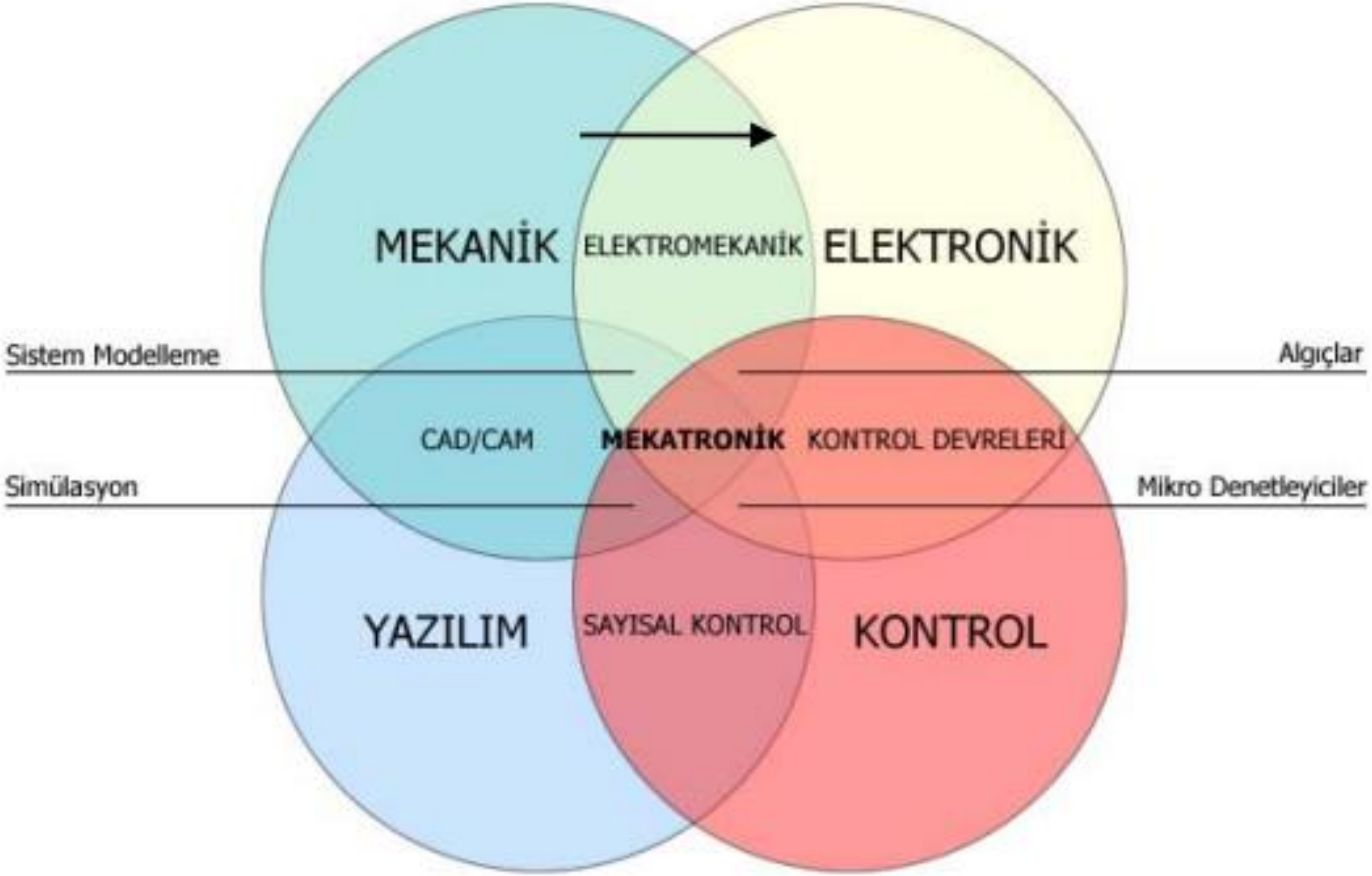
Ayrık Zamanda Durum Uzayı Denklemlerinin Çözümü

Aşağıdaki gibi N-boyutlu ayrık-zamanlı sistemi ele alalım:

$$\begin{aligned} \mathbf{q}[n+1] &= \mathbf{A}\mathbf{q}[n] + \mathbf{B}x[n] \\ y[n] &= \mathbf{C}\mathbf{q}[n] + \mathbf{D}x[n] \end{aligned}$$

burada $\mathbf{A}, \mathbf{B}, \mathbf{C}$ ve, $\mathbf{D}$ matrislerinin boyutları sırasıyla, $N \times N, N \times 1, 1 \times N$ ve 1×1 şeklindedir. Buradaki durum denklemlerinin çözümü olan $\mathbf{q}[n]$ vektörünün analitik ifadesinin bulunması için zaman domeni yaklaşımı ve z-dönüşümü yaklaşımı ele alınacaktır.

Mekatronik disiplinini oluşturan bilim dallarının birbiriyle etkileşimi Şekil 6'daki şemada verildiği gibi özetlenebilir. Buna göre, yukarıda bahsedilen alanlardan mekaniğin elektronik ve yazılımla arakesitleri sırasıyla elektromekanik ve CAD/CAM (bilgisayar destekli tasarım/bilgisayar destekli üretim) iken kontrol bileşeninin elektronik ve yazılımla ortak kümesi sırasıyla kontrol devreleri ve sayısal kontrol alanlarıdır. Ayrıca, şekilde işaret edilen sistem modelleme; mekanik, yazılım ve elektroniğin, simülasyon (benzetim); mekanik, yazılım ve kontrolün, mikro denetleyiciler; elektronik, kontrol ve yazılımın ve nihayet algıçlar (algılayıcılar) da mekanik, elektronik ve kontrol alanlarının ortak ürünleridir. İfade edilen dört ana disiplinin kesişimi de mekatroniği oluşturmaktadır [1].

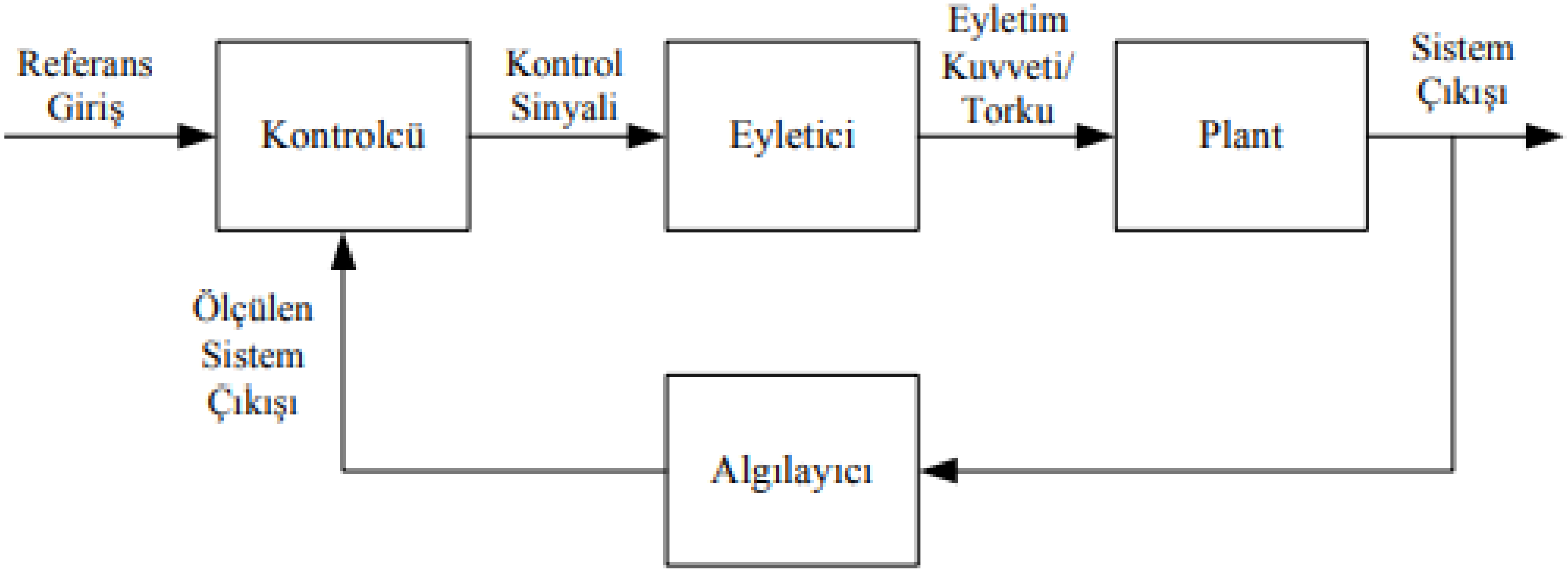


Mekatroniği oluşturan bilim dallarının birbiriyle etkileşimi [1]

Mekatroniğin uygulama alanları

- Modelleme
- Kontrol sistemleri
- Endüstriyel otomasyon (Barkod sistemleri, üretim bandı,..)
- Bina otomasyonu (Güvenlik, iklimlendirme, …)
- Sistem entegrasyonu
- Akıllı kontrol
- Robotik
- Mikroelektronik ve optoelektronik devreler
- Tıbbi uygulamalar
- Havacılık mühendisliği (otomatik pilotlar, insansız hava araçları)
- Otomotiv sistemleri
- …

Kontrol mühendisliği açısından bakıldığında, mekatronik sistemler; kontrol edilen sistemi (plantı) mekanik yapıda olan ve genel yapısı Şekil 7'deki gibi tanımlanabilen kapalı çevrim kontrol sistemleridir. En basit yapıdaki tek giriş ve tek çıkışlı uygulamaları göz önüne alınarak bir betimleme yapılırsa, kapalı çevrim kontrol sistemlerinde amaç; kullanıcı tarafından dışarıdan sağlanan ve belirlenen kontrol değişkeni için olması gereken değeri gösteren referans girişle kontrol değişkeninin (sistem çıkışının) bir algılayıcı tarafından ölçülen değerini kullanarak kontrolcü birimi tarafından üretilen kontrol sinyalini mekanik harekete (eyletim kuvveti veya torkuna) çeviren bir eyletici aracılığıyla kontrol değişkeni olarak seçilen plant parametresinin istenen değer veya değerlere getirilmesidir. Bahsedilen değer bir su tankının su seviyesi veya ele alınan bir kara aracının seyir hızı gibi sabit bir büyüklük ise kontrol sistemi bir "düzenleme sistemi (*İng. regulator system*)", zaman içerisinde genliği ve/veya yönü değişen bir değişkense de "takip sistemi (*İng. tracking system*)" olarak adlandırılmaktadır [9], [10].



Kapalı çevrim kontrol sistemi genel yapısı

Model Tabanlı Kontrol Yöntemleri

Mekatronik sistemlerde uygulanan model tabanlı kontrol (*İng. model-based control*) yöntemleri, temelde üç alt başlık altında ele alınabilir:

i. Klasik kontrol yöntemleri
ii. Modern kontrol yöntemleri
iii. Gürbüz kontrol yöntemleri

Klasik kontrol yöntemleri

Kontrol edilecek sistemin (plantın) tek giriş ve tek çıkışlı olması durumunda en fazla tercih edilen yaklaşım "klasik" olarak adlandırılan kontrol yöntemlerinin kullanılmasıdır. Klasik kontrol yöntemlerinde kontrol kuralı; kontrol sisteminin referans girişi ve ölçülen sistem çıkışı arasındaki işletim hatası (hata) baz alınarak hata, hata integrali ve hata türevi büyüklüklerinden uygun şekilde seçilenlerin, genliği plant dinamiği ve başarım isterleri doğrultusunda belirlenen kazanç katsayılarıyla çarpılmasıyla elde edilmektedir. Belirtilen yaklaşımda, kazançların belirlenmesi amacıyla gerekli cebirsel işlemlerin kolaylıkla yapılabilmesi ve ayrıca elde edilen nihai denklemlerin oluşturulan kontrol sisteminin frekans kümesindeki davranışını belirlemek üzere kullanılabilmesi amacıyla zaman kümesinde ifade edilen dinamik denklemlere Laplace dönüşümü uygulanmakta ve tasarımda hesaba katılacak eşitliklerin tamamı Laplace değişkeni ("s") cinsinden ifade edilmektedir. Elde edilen kontrol kuralı yalnızca hatanın hesaplanan bir kazançla çarpılması ile oluşturuluyorsa mevcut kontrol yöntemi "P (oransal) kontrol", hata ve hatanın integrali göz önüne alınıyorsa "PI (oransal ve integral) kontrol" ve hata ve hatanın türevi ele alınıyorsa "PD (oransal ve türevsel) kontrol" olarak adlandırılmakta olup, hata, hata türevi ve hata integralinin dikkate alındığı en genel klasik kontrol kuralı "PID (oransal, integral ve türevsel) kontrol" olarak tanımlanmaktadır. Bu anlamda P, PI ve PD kuralları PID kontrol kuralının türevleri olup, hata (*İng. error*) ve kontrol sinyali büyüklükleri sırasıyla E ve U harfleri ile ifade edilmek üzere, "s" değişkeni kullanılarak PID tipi bir kontrolcünün transfer fonksiyonu Şekil 9-(a)'daki gibi oluşturulabilir.

P, PI, PD,PID Kontrol Yöntemi

hata (*Ing. error*) ve kontrol sinyali büyüklükleri sırasıyla E ve U harfleri ile ifade edilmek üzere, Şekilde görülen K_p, K_i ve K_d sembolleri kontrolcünün sırasıyla oransal, integral ve türevsel kazançlarına karşılık gelmekte olup, PID tipi bir kontrolcünün transfer fonksiyonu

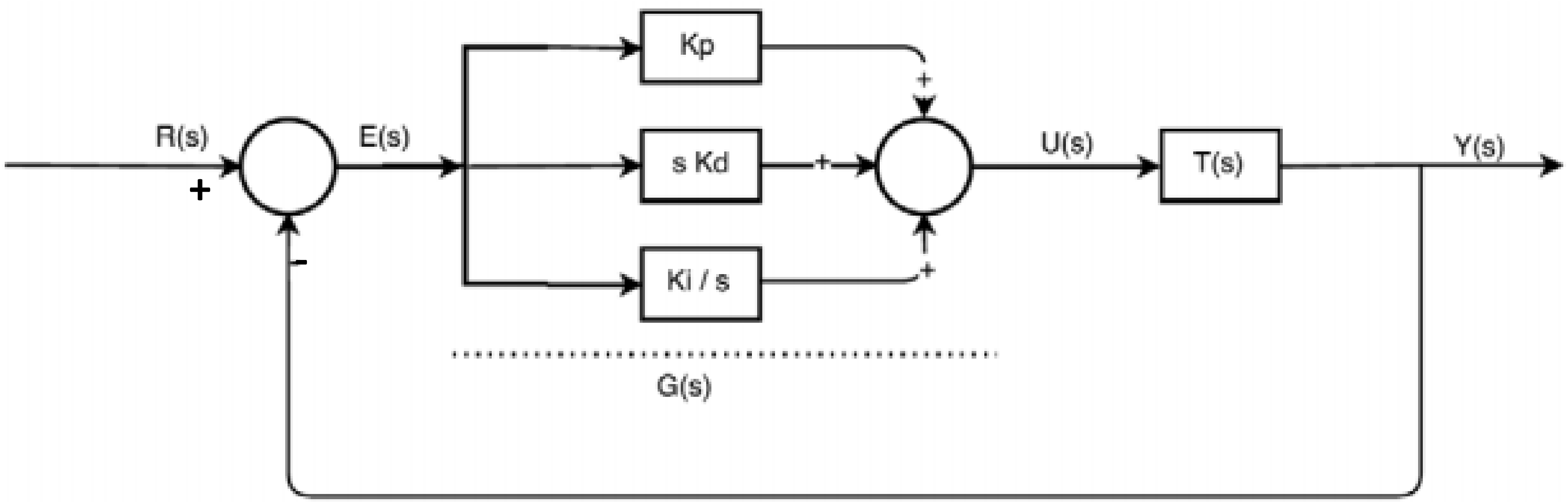
$$E(s) \longrightarrow \boxed{K_p + \frac{K_i}{s} + K_d\, s} \longrightarrow U(s)$$

PID kuralına göre çalışan kontrolcü yapısı

PID esaslı kontrolcüler, nispeten basit tasarımları, kontrolcü kazançlarının kolaylıkla ayarlanabilmesi ve pekçok sistem için yeterli düzeydeki gürbüzlükleri dolayısıyla, endüstriyel uygulamalarda en çok tercih edilen yapılardır. Özellikle kontrolcü olarak PLC (programlanabilir mantık kontrolcüsü) sistemlerinin kullanıldığı durumlar için, PID tipi kontrol vazgeçilmez hale gelmiştir.

PID Kontrol Yöntemi

PID Denetleyici Tasarımı



PID Denetleyicili Sistem Tasarımı

PID Kontrolör: Kontrol sisteminin geçici+kalıcı rejim kriterlerini (aşma-yerleşme-yükselme zamanı) + kalıcı durum hatası düzeltir. PID kontrolör,

$$u(t) = K_p e(t) + K_i \int e(t)dt + K_d \frac{d}{dt} e(t)$$

Projede PID denetleyicisi ile kontrol sinyali (u) üretmek için kullanacağınız formül

KPID vektörü aşağıdaki şekilde başlangıçta tanımlanmalıdır.

KPID = [Kp Kı Kd] ; (boyut:1x3 olmalı)

Jc = [(e(k) - e(k-1))   e(k)   (e(k) - 2*e(k-1) + e(k-2))]; (boyut: 3x1 olmalı)

u_new(k) = u_old(k) + KPID*Jc’;