



Dr. Sadık YILDIZ

sadikyildiz06@gmail.com

**Gazi Üniversitesi
Elektrik Elektronik Mühendisliği
(Teknoloji Fakültesi)**

2025– ANKARA

Sayısal Taşıyıcı Modülasyonu

- *Sayısal modülasyon: Dijital veriye göre taşıyıcı sinyalin özelliklerinin değiştirilmesi*
- Taşıyıcı: Genellikle sinüzoidal bir sinyal
- Modülasyon Türleri:
 - ASK (Amplitude Shift Keying)
 - FSK (Frequency Shift Keying)
 - PSK (Phase Shift Keying)
 - QPSK (Quadrature PSK)

Sayısal Taşıyıcı Modülasyonu

2. FSK – Frekans Kaydırmalı Anahtarlama (Frequency Shift Keying)

- FSK'de dijital veri bitleri (örneğin 0 ve 1), **iki farklı sinüzoidal taşıyıcı frekansı** ile temsil edilir.
 - Bit 0: düşük frekanslı taşıyıcı (f_0)
 - Bit 1: yüksek frekanslı taşıyıcı (f_1)
- **İkili FSK modülasyonu** (BFSK) için taşıyıcı sinyal şu şekilde tanımlanır:
$$s(t) = \begin{cases} A \cos(2\pi f_0 t) & , \text{ eğer } b(t) = 0 \\ A \cos(2\pi f_1 t) & , \text{ eğer } b(t) = 1 \end{cases}, 0 \leq t \leq T_b$$
- Bu denklemde;
- A: taşıyıcı genliği
- f_0 : bit 0 için taşıyıcı frekansı
- f_1 : bit 1 için taşıyıcı frekansı
- T_b : bit süresi
- $b(t)$: dijital bilgi biti (0 veya 1)

Sayısal Taşıyıcı Modülasyonu

2. FSK – Frekans Kaydırmalı Anahtarlama (Frequency Shift Keying)

➤ FSK Modülasyonunun Temel Özellikleri

- Frekans tabanlıdır: bilgi, genlik veya faz yerine taşıyıcı frekans ile iletilir.
- Spektrumda iki frekans bulunur: f_0 ve f_1
- Darbe benzeri geçişler yoktur, bu nedenle faz sürekliliği korunabilir (CPFSK gibi varyantlar ile).
- Gürültüye karşı ASK'e göre daha dayanıklıdır.
- FSK, sayısal bilgiyi farklı frekanslara sahip taşıyıcılarla kodlar.
- Bit 0 ve 1, sırasıyla f_0 ve f_1 frekanslı sinyallerle temsil edilir.
- Demodülasyon aşamasında genellikle karşılaştırmalı filtreleme veya ilinti alıcıları kullanılır.

Sayısal Taşıyıcı Modülasyonu

2. FSK – Frekans Kaydırmalı Anahtarlama (Frequency Shift Keying)

- **FSK Demodülasyonu**
- FSK (Frequency Shift Keying – Frekans Kaydırmalı Anahtarlama) modülasyonunun **demodülasyonu**, alınan sinyalin taşıdığı frekansa bakarak hangi bitin (0 mı 1 mi) gönderildiğinin belirlenmesidir.

Sayısal Taşıyıcı Modülasyonu

2. FSK – Frekans Kaydırmalı Anahtarlama (Frequency Shift Keying)

➤ FSK Demodülasyon Yöntemleri

1. Eşevreli (Coherent) Demodülasyon

Bu yöntemde, alıcı tarafında göndericiyle **faz ve frekans** olarak eşleşmiş (eşevreli) bir referans taşıyıcı üretilir.

Bu referans sinyallerle ilinti (korelasyon) alınarak karar verilir.

◆ Formül:

Alınan sinyal $r(t)$, taşıyıcı sinyallerle ilintilenir:

$$Z_0 = \int_0^{T_b} r(t) \cdot \cos(2\pi f_0 t) dt$$

$$Z_1 = \int_0^{T_b} r(t) \cdot \cos(2\pi f_1 t) dt$$

- Z_0 : Bit 0'a ait taşıyıcı frekans ile ilinti sonucu
- Z_1 : Bit 1'e ait taşıyıcı frekans ile ilinti sonucu

◆ Karar Kuralı:

$$\text{Eğer } Z_1 > Z_0 \Rightarrow \hat{b} = 1$$

$$\text{Eğer } Z_1 < Z_0 \Rightarrow \hat{b} = 0$$

Yani hangi frekans bileşeni daha baskınsa, o bitin gönderildiğine karar verilir.

Sayısal Taşıyıcı Modülasyonu

2. FSK – Frekans Kaydırmalı Anahtarlama (Frequency Shift Keying)

➤ FSK Demodülasyon Yöntemleri

2. Eşevresiz (Non-Coherent) Demodülasyon

Bu yöntemde taşıyıcı sinyallerle faz eşleştirmesi yapılmaz. Bunun yerine her frekansa özgü **bant geçiren filtre** (BPF) kullanılarak hangi frekansın daha güçlü olduğu belirlenir.

◆ Formül Yaklaşımı:

Filtre çıkışlarının enerjileri hesaplanır:

$$E_0 = \int_0^{T_b} [\text{BPF}_{f_0}\{r(t)\}]^2 dt$$

$$E_1 = \int_0^{T_b} [\text{BPF}_{f_1}\{r(t)\}]^2 dt$$

◆ Karar Kuralı:

$$\text{Eğer } E_1 > E_0 \Rightarrow \hat{b} = 1$$

$$\text{Eğer } E_1 < E_0 \Rightarrow \hat{b} = 0$$

Bu yöntemde, faz bilgisine ihtiyaç olmadığı için donanımsal olarak daha basittir ancak daha yüksek hata oranına sahiptir.

Sayısal Taşıyıcı Modülasyonu

2. FSK – Frekans Kaydırmalı Anahtarlama (Frequency Shift Keying)

➤ FSK Demodülasyon Yöntemleri

3. İlinti Alıcısı (Matched Filter veya Correlator)

Özellikle eşevreli demodülasyon için kullanılır. Her frekansa karşılık gelen taşıyıcı ile alınan sinyal çarpılarak bir integral (veya toplam) alınır. Bu toplam, gelen bitin hangi frekans bileşenine daha yakın olduğunu ortaya çıkarır.

◆ Diskret Zaman Formu (sayısal uygulamalar için):

$$Z_0 = \sum_{n=0}^{N-1} r[n] \cdot \cos\left(2\pi f_0 \frac{n}{f_s}\right)$$

$$Z_1 = \sum_{n=0}^{N-1} r[n] \cdot \cos\left(2\pi f_1 \frac{n}{f_s}\right)$$

- $r[n]$: alınan sinyalin örneklenmiş hali
- f_s : örnekleme frekansı
- N : bir bit süresi boyunca örnek sayısı

Sayısal Taşıyıcı Modülasyonu

2. FSK – Frekans Kaydırmalı Anahtarlama (Frequency Shift Keying)

➤ *FSK Demodülasyon Yöntemleri*

Karar Eşiği ve Gürültü Etkisi

Gürültü ortamında karar verirken genellikle bir eşik değeri kullanılır. Eşit olasılıkla bit gönderimi varsayımı altında **optimal karar eşiği**, iki taşıyıcıya karşılık gelen ortalama enerji değerlerinin ortasıdır.

Örnek:

$$\text{Karar eşiği} = \frac{Z_0 + Z_1}{2}$$

Sayısal Taşıyıcı Modülasyonu

2. FSK – Frekans Kaydırmalı Anahtarlama (Frequency Shift Keying)

➤ FSK Demodülasyon Yöntemleri

Yöntem	Referans Gerekir	Faz Uyumu Gerekir	Gürültüye Dayanıklılık	Donanım Karmaşıklığı
Eşevreli	Evet	Evet	Yüksek	Yüksek
Eşevresiz	Hayır	Hayır	Orta	Düşük
İlinti (uyumlu)	Evet	Evet	Yüksek	Orta-Yüksek

Sayısal Taşıyıcı Modülasyonu

2. FSK Modülasyonu

```

1 - clc; clear; close all;
2
3 % Parametreler
4 - fc0 = 2000; % Bit 0 için taşıyıcı frekans
5 - fc1 = 4000; % Bit 1 için taşıyıcı frekans
6 - fs = 100000; % Örnekleme frekansı
7 - Tb = 0.01; % Bit süresi
8 - t = 0:1/fs:Tb-1/fs; % Her bit için zaman vektörü
9
10 - data = [0 1 1 0 1]; % Bilgi verisi
11 - bitstream = []; % Bilgi sinyalini genlik olarak göstermek için
12 - fsk_signal = []; % Modüleli sinyal
13
14 % Modülasyon işlemi
15 - for i = 1:length(data)
16 -     bit_val = data(i)*ones(1, length(t)); % 0 veya 1 düzeyinde bilgi sinyali
17 -     bitstream = [bitstream bit_val]; % Bilgi sinyali birleştiriliyor
18
19 -     if data(i) == 0
20 -         carrier = cos(2*pi*fc0*t); % fc0 taşıyıcısı
21 -     else
22 -         carrier = cos(2*pi*fc1*t); % fc1 taşıyıcısı
23 -     end
24
25 -     fsk_signal = [fsk_signal carrier]; % Modüleli sinyal
26 - end
27
28 % Genel zaman eksenini
29 - T = 0:1/fs:Tb*length(data)-1/fs;
30
31 % Taşıyıcı sinyallerin örnekleri (tek bitlik)
32 - carrier0 = cos(2*pi*fc0*t);
33 - carrier1 = cos(2*pi*fc1*t);

```

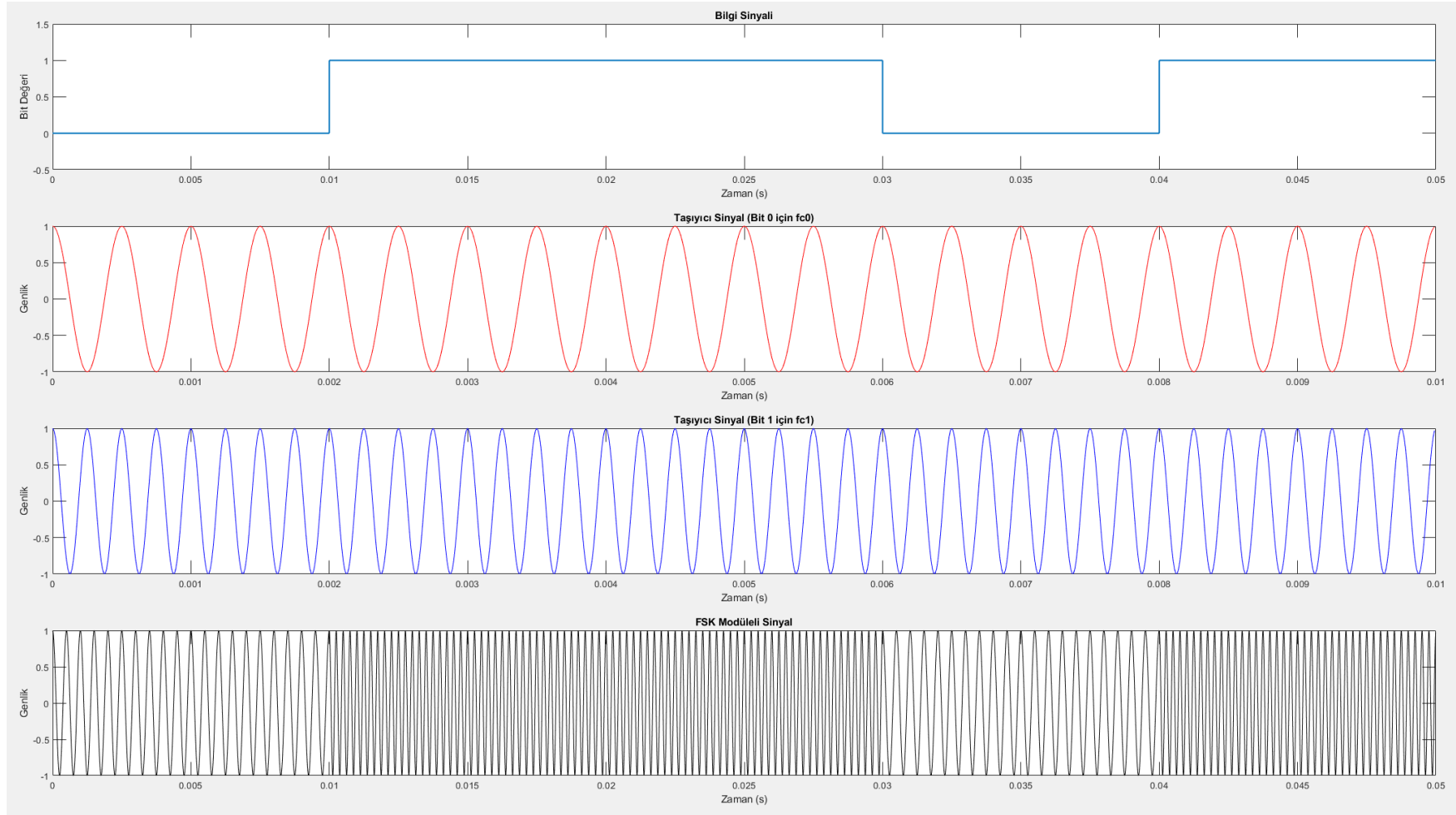
```

31 % Taşıyıcı sinyallerin örnekleri (tek bitlik)
32 - carrier0 = cos(2*pi*fc0*t);
33 - carrier1 = cos(2*pi*fc1*t);
34
35 % Grafik çizimi
36 - figure;
37 - subplot(4,1,1)
38 - stairs(T, bitstream, 'LineWidth', 1.5)
39 - ylim([-0.5 1.5])
40 - title('Bilgi Sinyali')
41 - xlabel('Zaman (s)')
42 - ylabel('Bit Değeri')
43
44 - subplot(4,1,2)
45 - plot(t, carrier0, 'r')
46 - title('Taşıyıcı Sinyal (Bit 0 için fc0)')
47 - xlabel('Zaman (s)')
48 - ylabel('Genlik')
49
50 - subplot(4,1,3)
51 - plot(t, carrier1, 'b')
52 - title('Taşıyıcı Sinyal (Bit 1 için fc1)')
53 - xlabel('Zaman (s)')
54 - ylabel('Genlik')
55
56 - subplot(4,1,4)
57 - plot(T, fsk_signal, 'k')
58 - title('FSK Modüleli Sinyal')
59 - xlabel('Zaman (s)')
60 - ylabel('Genlik')

```

Sayısal Taşıyıcı Modülasyonu

2. FSK Modülasyonu



Sayısal Taşıyıcı Modülasyonu

2. FSK Demodülasyonu

```

1 -   clc; clear; close all;
2
3   % Parametreler
4 -   fc0 = 2000;
5 -   fc1 = 4000;
6 -   fs = 100000;
7 -   Tb = 0.01;
8 -   t = 0:1/fs:Tb-1/fs;
9 -   N = length(t);
10
11  % Gelen sinyal (fc1 taşıyıcısı ile modüle edilmiş + gürültü eklenmiş)
12 -  r = cos(2*pi*fc1*t) + 0.5*randn(1, N); % Bit 1 gönderildiğini varsayıyoruz
13
14  % Referans taşıyıcılar
15 -  ref0 = cos(2*pi*fc0*t); % Bit 0 için referans taşıyıcı
16 -  ref1 = cos(2*pi*fc1*t); % Bit 1 için referans taşıyıcı
17
18  % İlinti sonuçları (korelasyon)
19 -  z0 = sum(r .* ref0);
20 -  z1 = sum(r .* ref1);
21
22  % Karar
23 -  if z1 > z0
24 -      karar = 'Bit 1 alınmıştır';
25 -  else
26 -      karar = 'Bit 0 alınmıştır';
27 -  end

```

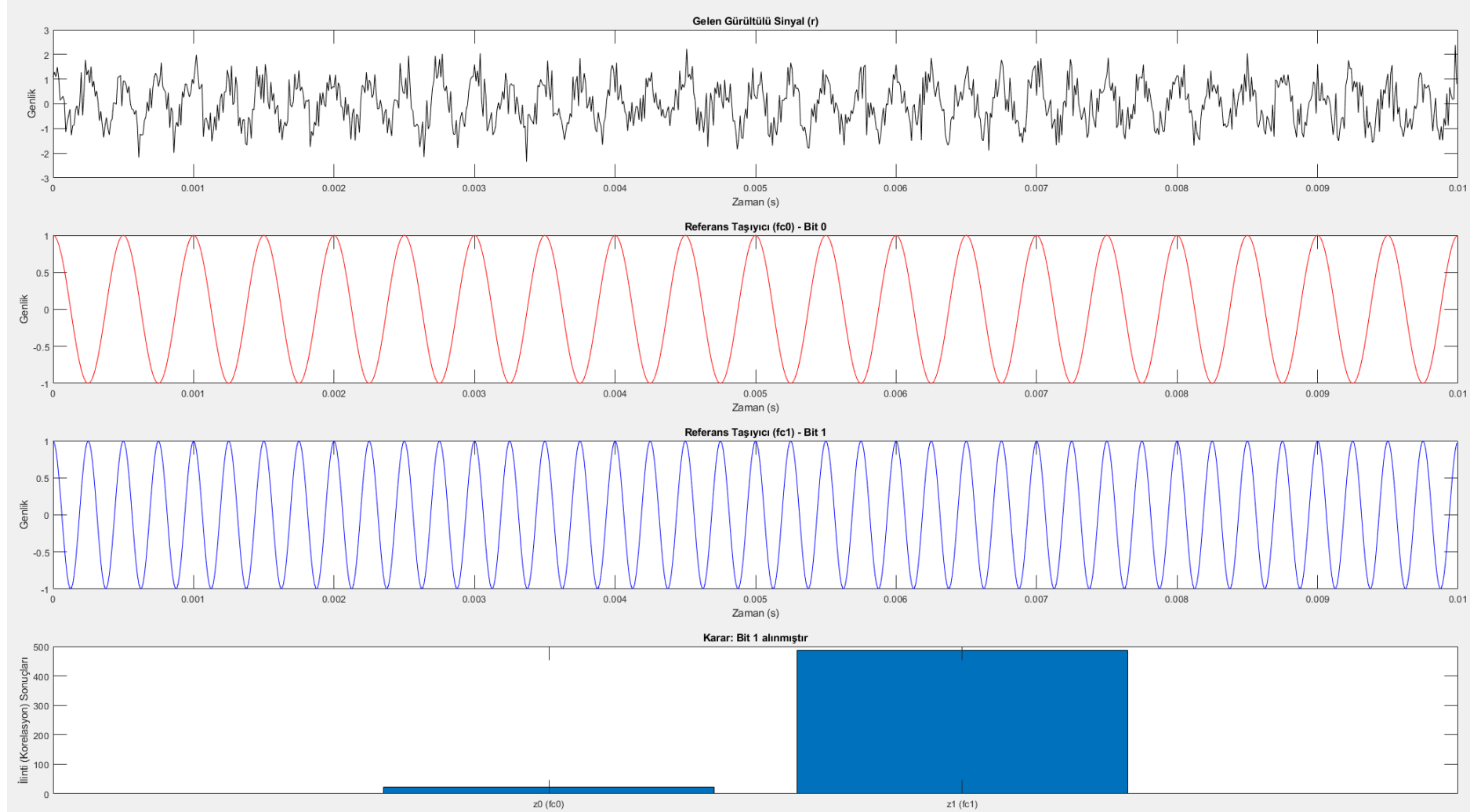
```

28
29  % Grafik çizimi
30 -  figure;
31
32 -  subplot(4,1,1)
33 -  plot(t, r, 'k')
34 -  title('Gelen Gürültülü Sinyal (r)')
35 -  xlabel('Zaman (s)')
36 -  ylabel('Genlik')
37
38 -  subplot(4,1,2)
39 -  plot(t, ref0, 'r')
40 -  title('Referans Taşıyıcı (fc0) - Bit 0')
41 -  xlabel('Zaman (s)')
42 -  ylabel('Genlik')
43
44 -  subplot(4,1,3)
45 -  plot(t, ref1, 'b')
46 -  title('Referans Taşıyıcı (fc1) - Bit 1')
47 -  xlabel('Zaman (s)')
48 -  ylabel('Genlik')
49
50 -  subplot(4,1,4)
51 -  bar([0 1], [z0 z1])
52 -  xticks([0 1])
53 -  xticklabels({'z0 (fc0)', 'z1 (fc1)'})
54 -  ylabel('İlinti (Korelasyon) Sonuçları')
55 -  title(['Karar: ', karar])

```

Sayısal Taşıyıcı Modülasyonu

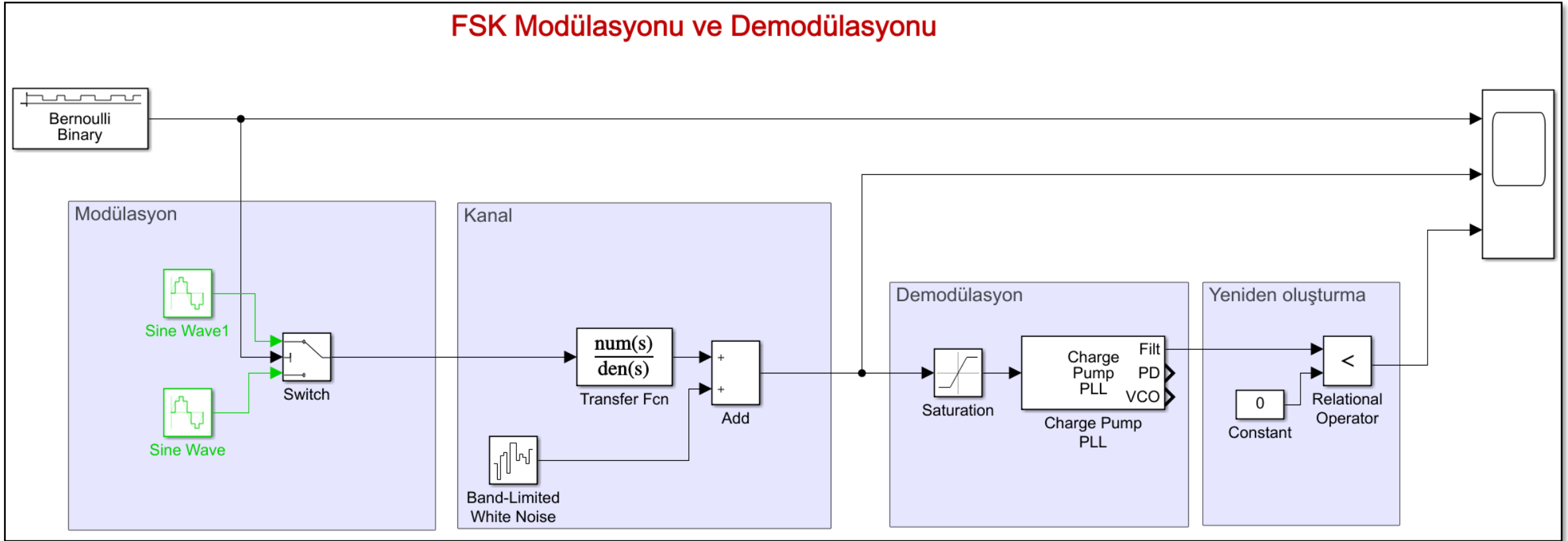
2. FSK Demodülasyonu



Sayısal Taşıyıcı Modülasyonu

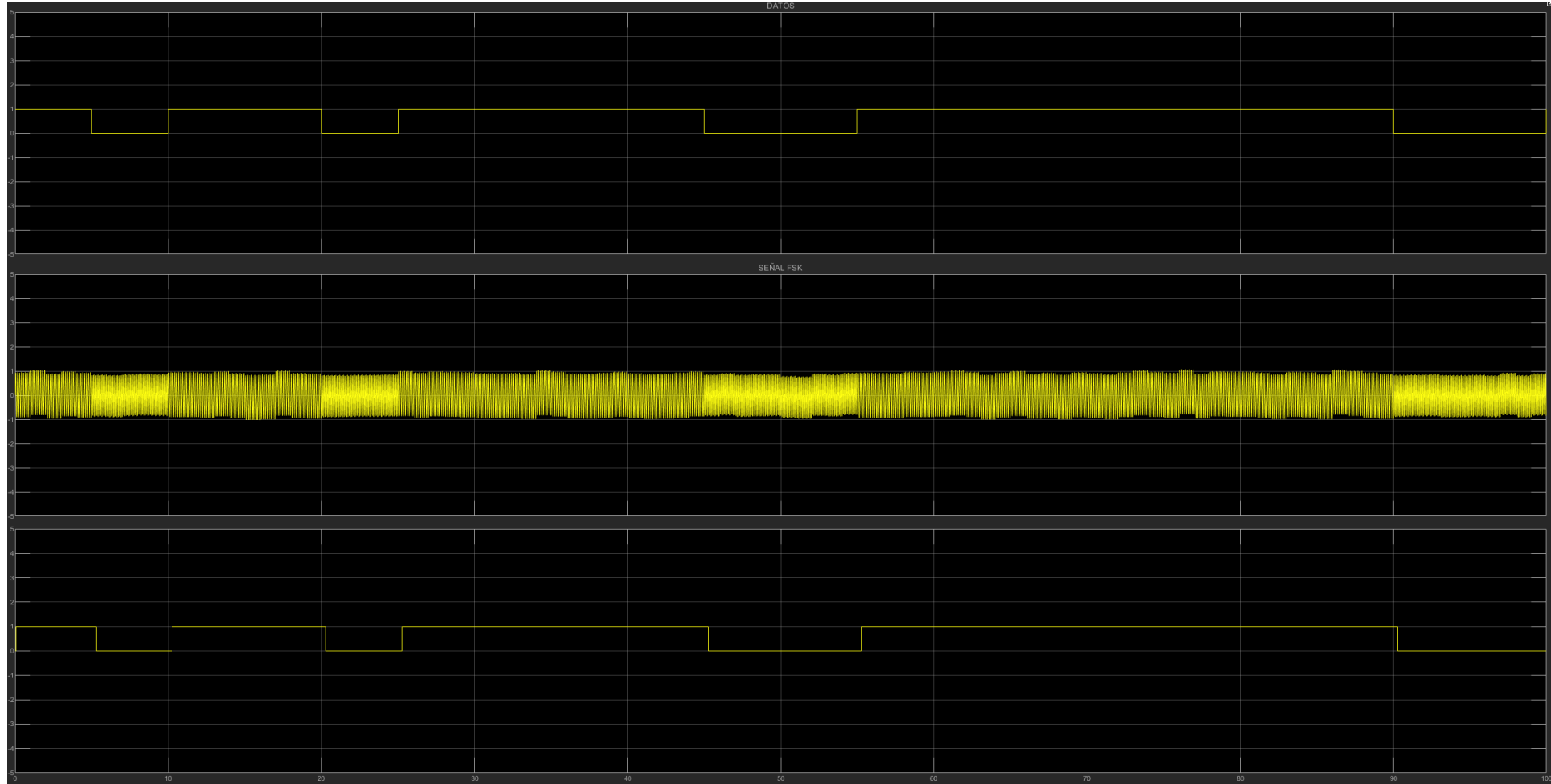
2. FSK Modülasyonu ve Demodülasyonu

FSK Modülasyonu ve Demodülasyonu



Sayısal Taşıyıcı Modülasyonu

2. FSK Modülasyonu ve Demodülasyonu



Sayısal Taşıyıcı Modülasyonu

2. FSK Modülasyonu ve Demodülasyonu

Block Parameters: Bernoulli Binary Generator

Bernoulli Binary Generator (mask) (link)

Generate a Bernoulli random binary number. To generate a vector output, specify the probability as a vector.

Parameters

Probability of a zero: 0.3

Initial seed: 61

Sample time: 5

☐ Frame-based outputs

☐ Interpret vector parameters as 1-D

Output data type: double

OK Cancel Help Apply

Block Parameters: Sine Wave1

Sine Wave

Output a sine wave:

$$O(t) = \text{Amp} * \sin(\text{Freq} * t + \text{Phase}) + \text{Bias}$$

Sine type determines the computational technique used. The parameters in the two types are related through:

Samples per period = $2 * \pi / (\text{Frequency} * \text{Sample time})$

Number of offset samples = $\text{Phase} * \text{Samples per period} / (2 * \pi)$

Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.

Parameters

Sine type: Time based

Time (t): Use simulation time

Amplitude: 1

Bias: 0

Frequency (rad/sec): $2 * \pi * 8$

Phase (rad): 0

Sample time: 0.01

☒ Interpret vector parameters as 1-D

OK Cancel Help Apply

Block Parameters: Sine Wave

Sine Wave

Output a sine wave:

$$O(t) = \text{Amp} * \sin(\text{Freq} * t + \text{Phase}) + \text{Bias}$$

Sine type determines the computational technique used. The parameters in the two types are related through:

Samples per period = $2 * \pi / (\text{Frequency} * \text{Sample time})$

Number of offset samples = $\text{Phase} * \text{Samples per period} / (2 * \pi)$

Use the sample-based sine type if numerical problems due to running for large times (e.g. overflow in absolute time) occur.

Parameters

Sine type: Time based

Time (t): Use simulation time

Amplitude: 1

Bias: 0

Frequency (rad/sec): $2 * \pi * 12$

Phase (rad): 0

Sample time: 0.01

☒ Interpret vector parameters as 1-D

OK Cancel Help Apply

Block Parameters: Switch

Switch

Pass through input 1 when input 2 satisfies the selected criterion; otherwise, pass through input 3. The inputs are numbered top to bottom (or left to right). The first and third input ports are data ports, and the second input port is the control port. The criteria for control port 2 are $u2 \geq \text{Threshold}$, $u2 > \text{Threshold}$ or $u2 \sim 0$.

Main Signal Attributes

Criteria for passing first input: $u2 \geq \text{Threshold}$

Threshold: 0.5

☒ Enable zero-crossing detection

OK Cancel Help Apply

Sayısal Taşıyıcı Modülasyonu

2. FSK Modülasyonu ve Demodülasyonu

Block Parameters: Transfer Fcn

Transfer Fcn

The numerator coefficient can be a vector or matrix expression. The denominator coefficient must be a vector. The output width equals the number of rows in the numerator coefficient. You should specify the coefficients in descending order of powers of s.

Parameters

Numerator coefficients:

[2*pi*20]

Denominator coefficients:

[1 2*pi*20]

Absolute tolerance:

auto

State Name: (e.g., 'position')

"

OK Cancel Help Apply

Block Parameters: Band-Limited White Noise

Band-Limited White Noise. (mask) (link)

The Band-Limited White Noise block generates normally distributed random numbers that are suitable for use in continuous or hybrid systems.

Parameters

Noise power:

[0.002]

Sample time:

1

Seed:

[23341]

☒ Interpret vector parameters as 1-D

OK Cancel Help Apply

Block Parameters: Saturation

Saturation

Limit input signal to the upper and lower saturation values.

Main Signal Attributes

Upper limit:

0.5

Lower limit:

-0.5

☒ Treat as gain when linearizing

☒ Enable zero-crossing detection

OK Cancel Help Apply

Sayısal Taşıyıcı Modülasyonu

2. FSK Modülasyonu ve Demodülasyonu

Block Parameters: Charge Pump PLL

Charge Pump PLL (mask) (link)

Implement a charge pump phase-locked loop using a digital phase detector. The three outputs are the outputs of the lowpass filter, the phase detector, and the voltage controlled oscillator (VCO). The input must be a sample-based scalar signal.

Parameters

Lowpass filter numerator:
[2*pi*1]

Lowpass filter denominator:
[1 2*pi*1]

VCO input sensitivity (Hz/V):
4

VCO quiescent frequency (Hz):
10

VCO initial phase (rad):
0

VCO output amplitude (V):
1

OK Cancel Help Apply