

PERCOBAAN 6

KODE KONVOLUSI HARD DAN SOFT DICISION

1. Tujuan :

Setelah melakukan praktikum, mahasiswa diharapkan dapat :

- Membuat simulasi dengan kode konvolusi menggunakan rangkaian encoder yang berbeda-beda.
- Membandingkan hasil pengkodean kembali antara hard dicision dan soft dicision pada proses pengkodean kode konvolusi
- Menghitung sindrom pada codeword bernoise

2. Prosedur Percobaan :

A. Pengkodean Tanpa Error

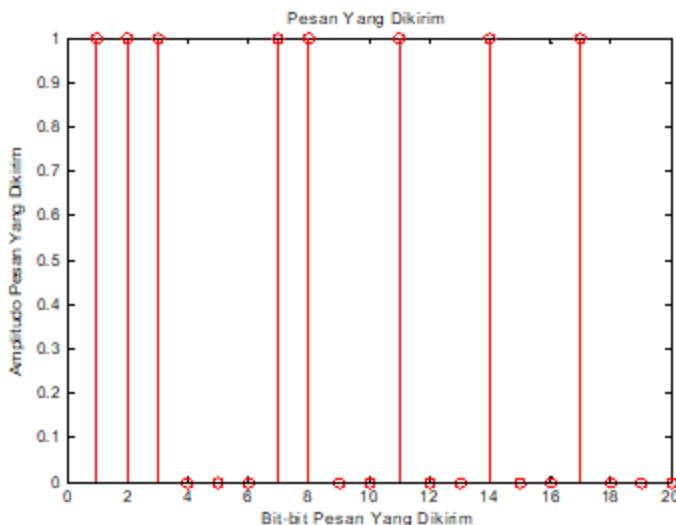
1. Masukan = Pesan = Informasi

Tuliskan pesan yang akan dikirim sebanyak 20 bit :

```
>> pesan_kirim=randint(20,1)
pesan_kirim =
1
0
0
0
1
0
1
0
0
1
1
0
0
1
0
0
1
1
1
1
```

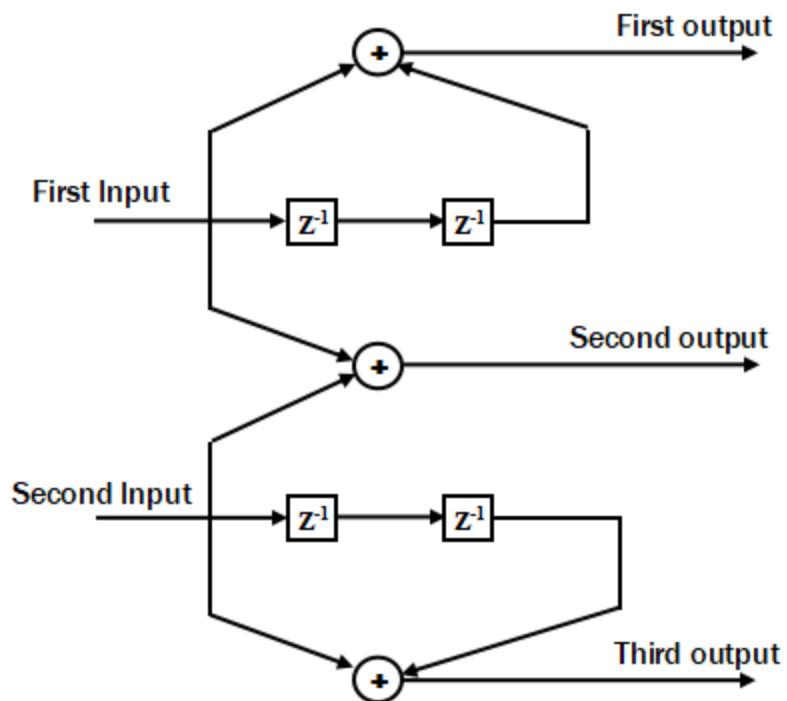
Tampilkan Pesan yang dikirim :

```
>> stem(pesan_kirim,'r')
>> title('Pesan Yang Dikirim')
>> xlabel('Bit-bit Pesan Yang Dikirim')
>> ylabel('Amplitudo Pesan Yang Dikirim')
```



2. Proses Pengkodean (Encoder)

Untuk proses pengkodean, digunakan metode pencarian trellis : rate yang digunakan 2/3 , struktur encoder seperti pada Gambar 1.



Gambar 1. encoder dengan rate 2/3

```

>> t=poly2trellis([3 3],[5 4 0;0 4 5]); % membangkitkan trellis dari encoder

t =
    numInputSymbols: 4
    numOutputSymbols: 8
    numStates: 16
    nextStates: [16x4 double]
    outputs: [16x4 double]

>> codeword = convenc(pesan_kirim,t)

codeword =
    1
    0
    1
    1
    1
    0
    1
    0
    1
    0
    0
    1
    0
    0
    0
    1
    1
    0
    1
    1
    1
    0
    0
    1
    1
    1
    0
    0
    1
    1
    1
    0
    0
    0
    0

```

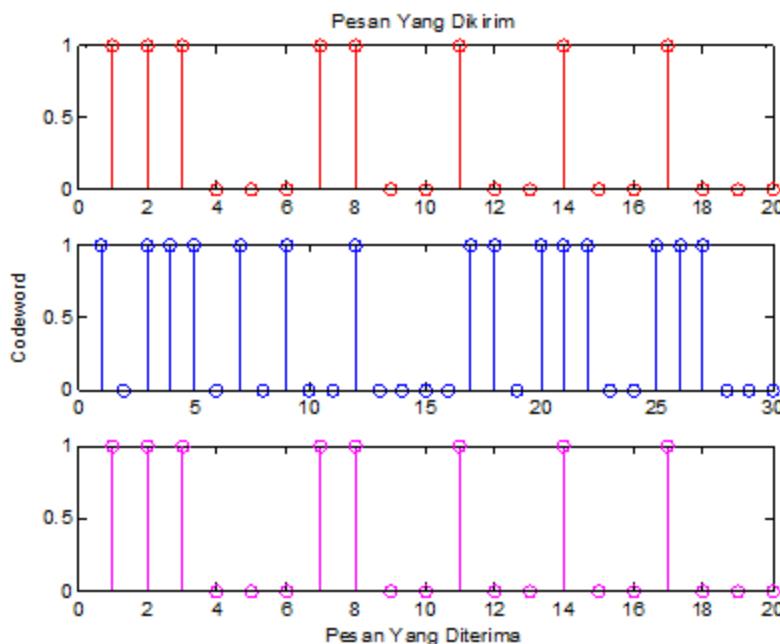
3. Dekoder (Pengkodean Kembali)

Pengkodean kembali pada kode konvolusi dapat dilakukan dengan menggunakan dua decision yaitu "hard" decision dan "soft" decision.

```
Pengkodean kembali menggunakan hard decision :  
>> tb=2;  
>> pesan_terima=vitdec(codeword,t,b,'trunc','hard')  
pesan_terima =  
1  
1  
1  
0  
0  
0  
1  
1  
0  
0  
0  
1  
0  
0  
1  
0  
0  
0  
0  
>> cek=[pesan_kirim pesan_terima]  
cek =  
1 1  
1 1  
1 1  
0 0  
0 0  
0 0  
1 1  
1 1  
0 0  
0 0  
1 1  
0 0  
0 0  
1 1  
0 0  
0 0  
1 1  
0 0  
0 0  
0 0
```

Tampilkan bersama-sama antara pesan yang dikirim, codeword dan pesan yang diterima dalam satu figure, seperti pada gambar dibawah ini.

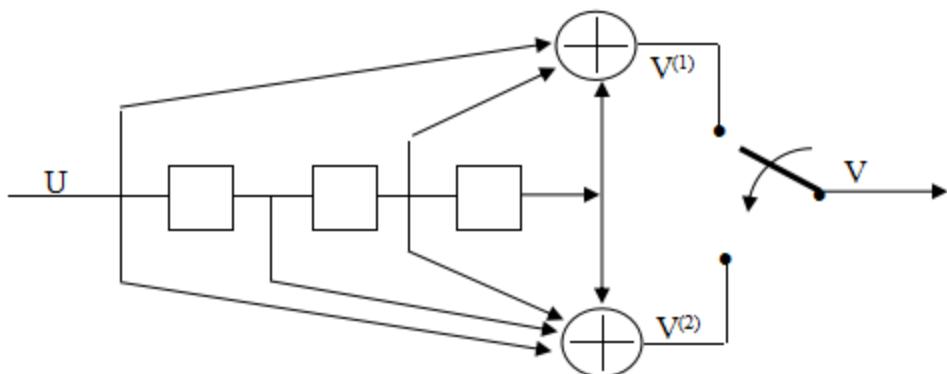
```
>> subplot(3,1,1)  
>> stem(pesan_kirim,'r');  
>> title('Pes an Yang Dikirim')  
>> subplot(3,1,2)  
>> stem(codeword,'b')  
>> ylabel('Codeword')  
>> subplot(3,1,3)  
>> stem(pesan_terima,'m')  
>> xlabel('Pes an Yang Diterima')
```



```
>> [number, ratio] = biterr(pesan_kirim, pesan_terima)  
number =  
0  
ratio =  
0
```

B. Pengkodean Dengan Error

1. Mengkodekan informasi menjadi sebuah codeword menggunakan encoder seperti pada gambar 2.



Gambar 2. encoder dengan rate 1/2

```

>> info_kirim=randint(20,1);
>> trellis=poly2trellis(4,[11 15]);
>> ns=64;%number state dari trellis
>> tb=2
>> codeword = convenc(info_kirim,trellis); % Encode the data.

```

2. Menambahkan noise pada codeword

```

>> N=length(codeword);
>> error=zeros(N,1);
>> error([5,11,13],1)=1
error =
0
0
0
0
1
0
0
0
%Penambahan noise pada codeword
>> code_error=xor(error,codeword')
>> noise_code=fix(code_error)
% Quantize to prepare for soft-decision decoding.
>> quan_code = quantiz(noise_code,[0.001,.1,.3,.5,.7,.9,.999]);%

```

(level kuantisasi -1) = $2^n - 1$

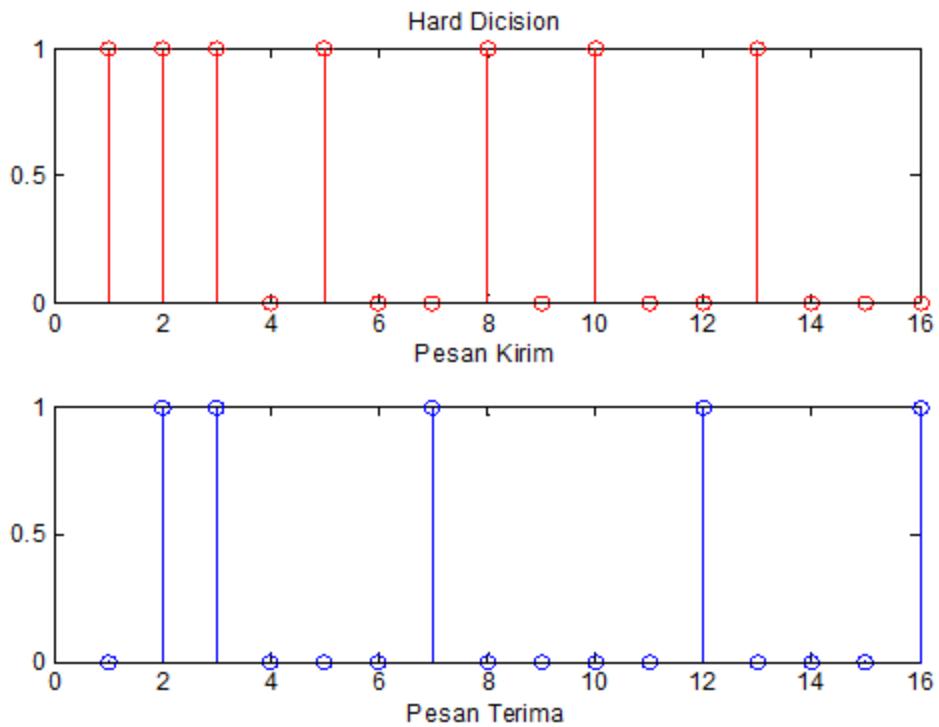
3. Mengkodekan kembali codeword menjadi sebuah informasi asal menggunakan Viterbi “hard dicision” dan “soft dicision”.

```
>> infoterima_hard = vitdec(noise_code,trellis,tb,'trunc','hard'); % Decode_hard.  
  
>> infoterima_soft = vitdec(quan_code,trellis,tb,'cont','soft',11); % Decode_soft.  
% Compute bit error rate.  
>> [numberhard,ratiohard] = biterr(infoterima_hard (delay+1:end),msg(1:end-delay))  
numberhard =  
8  
ratiohard =  
0.5000  
  
>> [numbersoft,ratiosoft] = biterr(infoterima_soft (delay+1:end),msg(1:end-delay))  
numbersoft =  
0  
ratiosoft =  
0  
  
>> infoterima_hard = infoterima_hard (delay+1:end);  
>> info_kirim=info_kirim (1:end-delay);  
>> infoterima_soft = infoterima_soft (delay+1:end);  
>> info_kirim=info_kirim (1:end-delay);
```

Index
kuantisasi

4. Plot hasil pengkodean

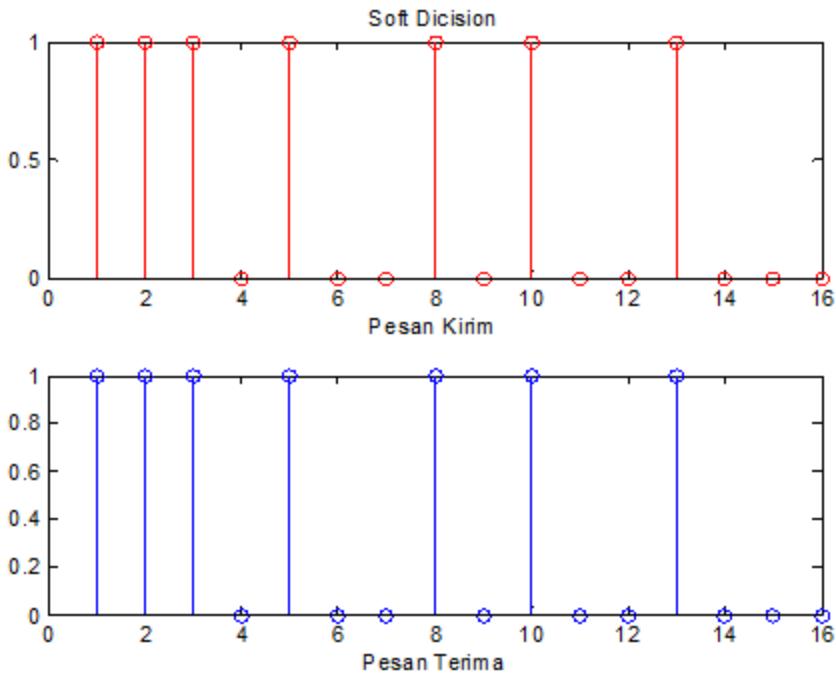
```
>> subplot(2,1,1)  
>> stem(info_kirim,'r');  
>> title('Soft Dicision')  
>> xlabel('Pesan Kirim')  
>> subplot(2,1,2)  
>> stem(infoterima_hard,'b')  
>> xlabel('Pesan Terima')
```



```

>> subplot(2,1,1)
>> stem(info_kirim,'r');
>> title('Hard Decision')
>> xlabel('Pesan Kirim')
>> subplot(2,1,2)
>> stem(infoterima_soft,'b')
>> xlabel('Pesan Terima')

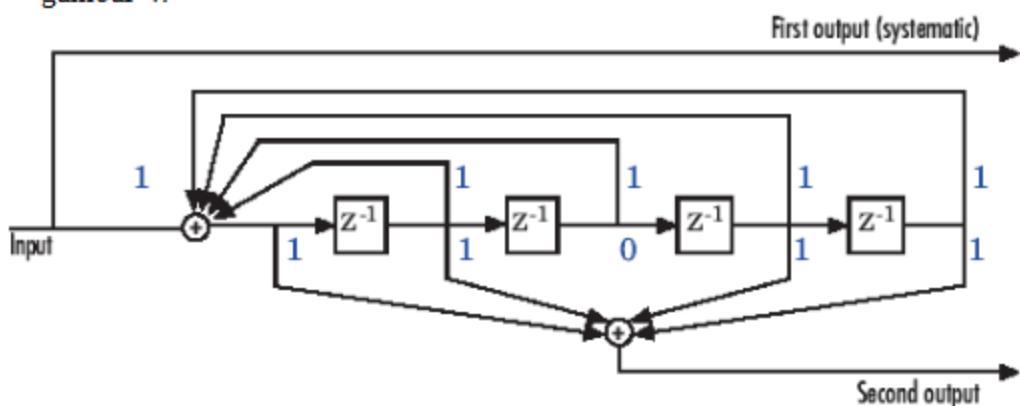
```



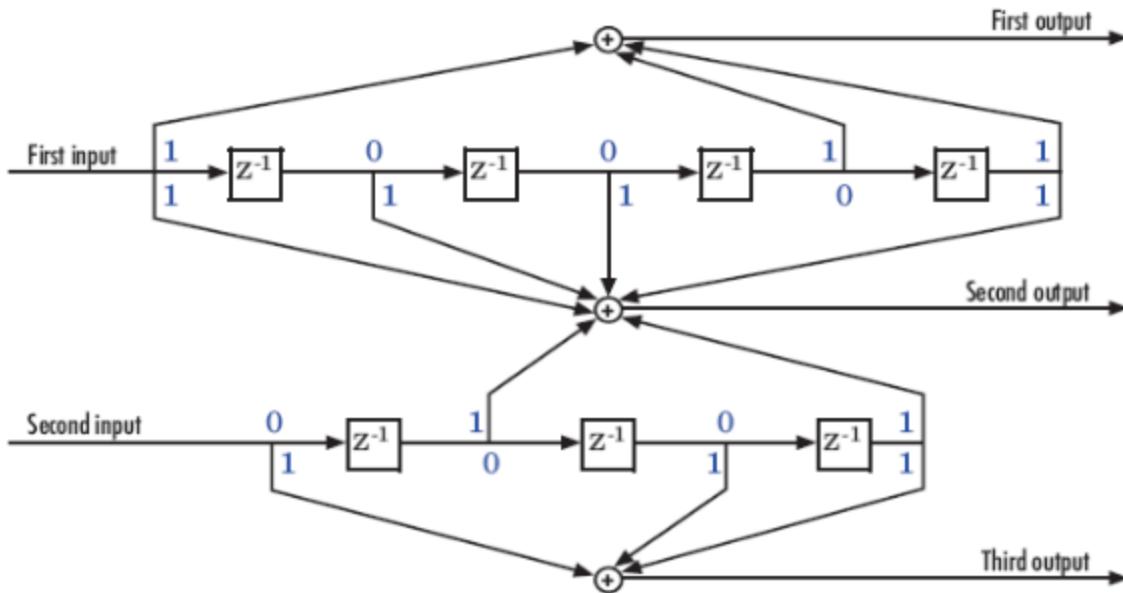
Tugas dalam praktikum :

A. Pesan sama, encoder berbeda decoder sama "hard dicision" (tanpa error) :

1. Buatlah program dalam satu m-file untuk mengkodekan sebuah informasi menggunakan kode konvolusi, dengan ketentuan sebagai berikut :
2. Bangkitkan informasi secara random sebanyak 20 bit.
3. Lakukan pengkodean (encoding) terhadap informasi tersebut menggunakan metode trellis dengan masing-masing rangkaian encoder seperti gambar 3. dan gambar 4.



Gambar 3. Encoder kode konvolusi dengan rate $\frac{1}{2}$



Gambar 4. Encoder kode konvolusi dengan rate 2/3

4. Kodekan kembali masing-masing codeword yang telah diperoleh dari point 3 dengan decoder “Hard dicision” . Kemudian bandingkan hasil dari keduanya.
 5. Plot semua hasil simulasi yang diperlukan untuk analisa.
- B. Pesan sama, encoder sama, decoder berbeda yaitu “hard dicision” dan “soft dicision” (ditambah dengan error) :
1. Buatlah program dalam satu m-file untuk mengkodekan sebuah informasi menggunakan kode konvolusi, dengan ketentuan sebagai berikut :
 2. Bangkitkan informasi secara random sebanyak 20 bit.
 3. Lakukan pengkodean (encoding) terhadap informasi tersebut menggunakan metode trellis dengan masing rangkaian encoder seperti gambar 1.
 4. Kodekan kembali masing-masing codeword yang telah diberi noise, dengan decoder “Hard dicision” dan “Soft dicision”. Kemudian bandingkan hasil dari keduanya.
 6. Plot semua hasil simulasi yang diperlukan untuk analisa.

---ooo0ooo---