LAMPIRAN

KODE PROGRAM SISTEM MIMO MENGGUNAKAN

MATLAB 7.0.1.

```
nsymb = 100000;
modtype = 'psk'; %nsymb dan tipe modulasi dapat
diubah sesuai keinginan

Mary = 2;
dobiterr = 'no';
snr = 0:2:14;

mc = 10;

for i_mc = 1:mc
    for i_s = 1:length(snr)
        coment = sprintf('monte-carlo = %d, snr
= %d', i_mc, snr(i_s));
        disp(coment);
        % Kanal Fading
        err1(i_mc,i_s)=
mrc(snr(i_s),nsymb,Mary,modtype,1,dobiterr);
        err4(i_mc,i_s)=
mrc(snr(i_s),nsymb,Mary,modtype,4,dobiterr);
        err8(i_mc,i_s) =
mrc(snr(i_s),nsymb,Mary,modtype,8,dobiterr);
        % Kanal AWGN
        errA(i_mc,i_s) =
kawgn(snr(i_s),nsymb,Mary,modtype);
    end
end

BER1 = mean(err1)./nsymb;
BER4 = mean(err4)./nsymb;
BER8 = mean(err8)./nsymb;
BERA = mean(errA)./nsymb;
```

```
figure;
semilogy(snr,BER1,'r-+');
hold on;
semilogy(snr,BER4,'g-o');
semilogy(snr,BER8,'b-*');
semilogy(snr,BERA,'c-x');
grid on;
hold off;
title('SER vs SNR untuk Modulasi BPSK');
legend('fading M=1','fading M = 4', ...
       'fading M = 8', 'AWGN')
xlabel('SNR (dB)');ylabel('SER');


function [err] = kawgn(snr,nsymb,Mary,modtype);

%Generate Data
%Possible Symbol Set
Set=[0:Mary-1]';
Smap=dmodce(Set,1,1,modtype,Mary);
%Generate Symbols
symb=randsrc(nsymb,1,[0:Mary-1]);
%Encode
msg=dmodce(symb,1,1,modtype,Mary);
%Noise Stats
Eav=Smap'*Smap/Mary;
NF=10^(snr/10);
S=sqrt(Eav/(2*NF));

noise=S*(randn(nsymb,1) + i*randn(nsymb,1));

msgN = msg + noise;

msgdet = ddemodce(msgN,1,1,modtype,Mary);

err = symerr(symb,msgdet);
```

```
function
[err]=mrc(snr,nsymb,Mary,modtype,nRx,dobiterr);
%Routine that preforms simulations based on PSK
Mary modulation
%and using coding scheme G1 (note this means no
coding)

nTx=2;
%Generate Data
%Possible Symbol Set
Set=[0:Mary-1]';
Smap=dmodce(Set,1,1,modtype,Mary);
%Generate Symbols
symb=randsrc(nsymb,1,[0:Mary-1]);
%Encode
msg=dmodce(symb,1,1,modtype,Mary);
%Noise Stats
Eav=Smap'*Smap/Mary;
NF=10^(snr/10);
S=sqrt(nTx*Eav/(2*NF));
noise=S*(randn(nsymb,nRx) +
i*randn(nsymb,nRx));

H = sqrt(1/2).*(randn(nsymb,nRx) +
j*randn(nsymb,nRx));

for ja=1:nRx
rx(:,ja) = H(:,ja).*msg + noise(:,ja);
end

% Combiner
S(length(msg),1)=[0];
for re=1:nRx,
    S = S + rx(:,re).*conj(H(:,re));
end;
order = nTx*nRx;
%Preform maximum likelihood demodulation
if order ~= 1
Hnorm = sum((H.*conj(H))');
Hnorm = Hnorm'; % change to be a column matrix
```

```
else
Hnorm = H.*conj(H);
end

 % change to be a column matrix
for L=1:Mary
    map(:,L) = Smap(L)*ones(length(S),1);
    Con = (Hnorm - 1).* map(:,L) .*
conj(map(:,L));
    est_S(:,L)= real((S - map(:,L)).*conj(S-
map(:,L)) + Con);
end;
[A,B]=min(est_S');
Srec = (B - 1)'; % turn into a column
%compute error
switch dobiterr
    case 'no'
        err=symerr(Srec,symb);
    case 'yes'
        err= biterr(Srec,symb);
end
```