Lampiran A1 Program Utama

```
directory_utama;
    cd(dir_utama)
        Global_parameter
    cd(dir_PU)

    Lsnr        = 0;
    Usnr        = 18;
    nsymb       = 1e3;
    jeda        = 2;
    Mary        = jml_konstelasi;
    modtype     = modulasi;
    stbc        = tipe_stbc;
    nRx         = jml_antena;
    dobiterr    = kesalahan;
    pausex      = 0.0;

    tic
    fname =
sprintf('stbc_%s_%d%s_%dRx_%s',stbc,Mary,modtype,nRx,dobiterr);
    sigStr=['Matrik' stbc];
    err=ones(((Usnr-Lsnr)/jeda)+1,1);
    index=0;

    set(handles.activex1,'value',10);
    set(handles.text12,'string','input parameter');
    pause(pausex);
    for snr=Lsnr:jeda:Usnr,
        set(handles.activex1,'value',(10+(index*3.5)));
        set(handles.text12,'string',['SNR = ' num2str(snr) '
dB']);
        pause(pausex);
        index=index+1;
        switch dobiterr
            case 'no'
                abscissa = 'SNR / (dB)';
                ordinate = 'Symbol Error Probability';
            case 'yes',
                K=log2(Mary);
                snr=snr + 10*log10(K);
                abscissa = 'SNR / (dB)';
                ordinate = 'Bit Error Probability';
        end

        switch sigStr
            case 'MatrikG1'

[err]=MatrikG1(index,snr,nsymb,Mary,modtype,nRx,err);
                Rate = 1;
            case 'MatrikG2'
                nsymb=nsymb - mod(nsymb,2);

[err]=MatrikG2(index,snr,nsymb,Mary,modtype,nRx,err);
                Rate = 1;

case 'MatrikG3'
                nsymb=nsymb - mod(nsymb,4);
```

```
[err]=MatrikG3(index,snr,nsymb,Mary,modtype,nRx,err);
                Rate = 1/2;
            case 'MatrikG4'
                snr;
                nsymb=nsymb - mod(nsymb,4);

[err]=MatrikG4(index,snr,nsymb,Mary,modtype,nRx,err);
                Rate = 1/2;
            case 'MatrikH3'
                nsymb=nsymb - mod(nsymb,3);

[err]=MatrikH3(index,snr,nsymb,Mary,modtype,nRx,err);
                Rate = 3/4;
            case 'MatrikH4'
                nsymb=nsymb - mod(nsymb,3);

[err]=MatrikH4(index,snr,nsymb,Mary,modtype,nRx,err);
                Rate = 3/4;
        end
    end

    panjang_snr = Lsnr:jeda:Usnr;
    bit_error_rate  = err/nsymb;

    %Troughput
    [k l] = size(bit_error_rate);
    p=[];
    q =0;
    for q=1:k
        for r=1:l
            p(q,r)=(1-bit_error_rate(q,r))^1000*11;
        end
    end
    throughput_total = p;
    %h1 = figure;
     %   semilogy(panjang_snr,throughput_total,'ko-');
      %  xlabel('SNR (dB)');
       % ylabel('Troughput');
        %title(tle)
        %grid on;

    cd(folder_analisa)
        save(fname, 'err', 'snr')
        save(['Tr' fname], 'throughput_total', 'snr')
    cd(dir_PU)

    speff = Rate*log2(Mary);
    str = ['%d bps/Hz, %d antena Rx, Matrik ' stbc ', modulasi '
'%d ' modtype];
    tle = [sprintf(str,speff,nRx,Mary)];
    fprintf('Finished.\n')
```

Lampiran A2 Fnction Matrik G2

```
function [err]=MatrikG2(index,snr,nsymb,Mary,modtype,nRx,err);
    directory_utama;
    cd(dir_utama)
        Global_parameter
    cd(folder_PU)
    nTx=2;
    Set=[0:Mary-1]';
    Smap=dmodce(Set,1,1,modtype,Mary);
    symb=randsrc(nsymb,1,[0:Mary-1]);
    save('symb.mat','symb')
    %load('symb.mat');
    msg=dmodce(symb,1,1,modtype,Mary);
    Eav=Smap'*Smap/Mary;
    NF=10^(snr/10);
    S=sqrt(nTx*Eav/(2*NF));
    for k=1:nTx:nsymb,
        %clc;
        Sv=msg(k:k+nTx-1);
        G2 = round([ Sv(1)   Sv(2);
              -Sv(2)' Sv(1)']);
        H=(randn(nTx,nRx) + i*randn(nTx,nRx))/sqrt(2);

        if check_OFDM == 1
            %OFDM
            input_OFDM1 = G2*H;
            [v w] = size(input_OFDM1);
            panjang_sinyal_in = v;
            for y = 1:w
                input_OFDM = input_OFDM1(:,y);
                cd(folder_transceiver)
                    transceiver
                    noise = S*randn(1,size_fft);
                    sinyal_in_receiver = sinyal_out_transceiver +
noise;
                cd(folder_receiver)
                    receiver
                    rx(:,y) = sinyal_output_receiver;
                cd(folder_PU)
            end
        else
            noise=S*(randn(2,nRx) + i*randn(2,nRx));
            rx=G2*H + noise;
        end

        S1=0; S2=0; Hnorm=0;
        for j=1:nRx,
            S1 = S1 + rx(1,j)*H(1,j)' + rx(2,j)'*H(2,j);
            S2 = S2 + rx(1,j)*H(2,j)' - rx(2,j)'*H(1,j);
            Hnorm = Hnorm + H(:,j)'*H(:,j);
        end




for L=1:Mary
            Con = (-1 + Hnorm) * Smap(L) * Smap(L)';
```

```
            est_s1(L)= real( (S1 - Smap(L))*(S1 - Smap(L))' +
Con);
            est_s2(L)= real( (S2 - Smap(L))*(S2 - Smap(L))' +
Con);
        end
        [A,B]=min(est_s1);
        shat(k)=B-1;
        [A,B]=min(est_s2);
        shat(k+1)=B-1;
    end
    err(index)=sum(shat~=symb');
```

Lampiran A3 Function Matrik G3

```
function [err]=MatrikG3(index,snr,nsymb,Mary,modtype,nRx,err);
    directory_utama;
    cd(dir_utama)
        Global_parameter
    cd(folder_PU)

    nTx=3;
    Set=[0:Mary-1]';
    Smap=dmodce(Set,1,1,modtype,Mary);
    symb=randsrc(nsymb,1,[0:Mary-1]);
    msg=dmodce(symb,1,1,modtype,Mary);
    Eav=Smap'*Smap/Mary;
    NF=10^(snr/10);
    S=sqrt(nTx*Eav/(2*NF));
    for k=1:4:nsymb,
        Sv=msg(k:k+3);
        G3_1 = [Sv(1) Sv(2) Sv(3);
            -Sv(2) Sv(1) -Sv(4);
            -Sv(3) Sv(4) Sv(1);
            -Sv(4) -Sv(3) Sv(2)];
        G3= [G3_1; conj(G3_1)];
        H=(randn(nTx,nRx) + i*randn(nTx,nRx))/sqrt(2);

        if check_OFDM == 1
            %OFDM
            input_OFDM1 = G3*H;
            [v w] = size(input_OFDM1);
            panjang_sinyal_in = v;
            for y = 1:w
               input_OFDM = input_OFDM1(:,y);
               cd(folder_transceiver)
                  transceiver
                  noise = S*randn(1,size_fft);
                  sinyal_in_receiver = sinyal_out_transceiver +
noise;
               cd(folder_receiver)
                   receiver
                   rx(:,y) = sinyal_output_receiver;
               cd(folder_PU)
            end
        else
            noise=S*(randn(8,nRx) + i*randn(8,nRx));
            rx= G3*H + noise;
        end

        S1=0; S2=0; S3=0; S4=0; Hnorm=0;
        for j=1:nRx,
            S1 = S1 +rx(1,j)*H(1,j)' + rx(2,j)*H(2,j)' +
rx(3,j)*H(3,j)' ...
               +rx(5,j)'*H(1,j) + rx(6,j)'*H(2,j) +
rx(7,j)'*H(3,j);
            S2 = S2 +rx(1,j)*H(2,j)' - rx(2,j)*H(1,j)' +
rx(4,j)*H(3,j)' ...
               +rx(5,j)'*H(2,j) - rx(6,j)'*H(1,j) +
rx(8,j)'*H(3,j);
```

```
S3 = S3 +rx(1,j)*H(3,j)' - rx(3,j)*H(1,j)' - rx(4,j)*H(2,j)' ...
                +rx(5,j)'*H(3,j) - rx(7,j)'*H(1,j) -
rx(8,j)'*H(2,j);
            S4 = S4 -rx(2,j)*H(3,j)' + rx(3,j)*H(2,j)' -
rx(4,j)*H(1,j)' ...
                -rx(6,j)'*H(3,j) + rx(7,j)'*H(2,j) -
rx(8,j)'*H(1,j);
            Hnorm = Hnorm + H(:,j)'*H(:,j);
        end

        for L=1:Mary
            Con = (-1 + 2*Hnorm) * Smap(L) * Smap(L)';
            est_s1(L)= real( (S1 - Smap(L))*(S1 - Smap(L))' +
Con);
            est_s2(L)= real( (S2 - Smap(L))*(S2 - Smap(L))' +
Con);
            est_s3(L)= real( (S3 - Smap(L))*(S3 - Smap(L))' +
Con);
            est_s4(L)= real( (S4 - Smap(L))*(S4 - Smap(L))' +
Con);
        end
        [A,B]=min(est_s1);
        shat(k)=B-1;
        [A,B]=min(est_s2);
        shat(k+1)=B-1;
        [A,B]=min(est_s3);
        shat(k+2)=B-1;
        [A,B]=min(est_s4);
        shat(k+3)=B-1;
    end
    err(index)=sum(shat~=symb');
```

Lampiran A4 Funtion Matrik G4

```
function [err]=MatrikG4(index,snr,nsymb,Mary,modtype,nRx,err);
    directory_utama;
    cd(dir_utama)
        Global_parameter
    cd(folder_PU)

    nTx = 4;
    delay = 8;
    Set=[0:Mary-1]';
    Smap=dmodce(Set,1,1,modtype,Mary);
    symb=randsrc(nsymb,1,[0:Mary-1]);
    msg=dmodce(symb,1,1,modtype,Mary);
    Eav=Smap'*Smap/Mary;
    NF=10^(snr/10);
    S=sqrt(nTx*Eav/(2*NF));
    for k=1:4:nsymb,
        Sv=msg(k:k+3);
        G4_1 = [Sv(1) Sv(2) Sv(3) Sv(4);
               -Sv(2) Sv(1) -Sv(4) Sv(3);
               -Sv(3) Sv(4) Sv(1) -Sv(2);
               -Sv(4) -Sv(3) Sv(2) Sv(1)];
        G4= [G4_1; conj(G4_1)];
        H=(randn(nTx,nRx) + i*randn(nTx,nRx))/sqrt(2);

        if check_OFDM == 1
            %OFDM
            input_OFDM1 = G4*H;
            [v w] = size(input_OFDM1);
            panjang_sinyal_in = v;
            for y = 1:w
                input_OFDM = input_OFDM1(:,y);
                cd(folder_transceiver)
                    transceiver
                    noise = S*randn(1,size_fft);
                    sinyal_in_receiver = sinyal_out_transceiver +
noise;
                cd(folder_receiver)
                    receiver
                    rx(:,y) = sinyal_output_receiver;
                cd(folder_PU)
            end
        else
            noise=S*(randn(8,nRx) + i*randn(8,nRx));
            rx= G4*H + noise;
        end

        S1=0; S2=0; S3=0; S4=0; Hnorm=0;
        for j=1:nRx,
            S1 = S1 + rx(1,j)*H(1,j)' + rx(2,j)*H(2,j)' +
rx(3,j)*H(3,j)' + rx(4,j)*H(4,j)' ...
                    +rx(5,j)'*H(1,j) + rx(6,j)'*H(2,j) +
rx(7,j)'*H(3,j) + rx(8,j)'*H(4,j);
```

```
S2 = S2 + rx(1,j)*H(2,j)' - rx(2,j)*H(1,j)' - rx(3,j)*H(4,j)' +
rx(4,j)*H(3,j)' ...
                    +rx(5,j)'*H(2,j) - rx(6,j)'*H(1,j) -
rx(7,j)'*H(4,j) + rx(8,j)'*H(3,j);
            S3 = S3 + rx(1,j)*H(3,j)' + rx(2,j)*H(4,j)' -
rx(3,j)*H(1,j)' - rx(4,j)*H(2,j)' ...
                    +rx(5,j)'*H(3,j) + rx(6,j)'*H(4,j) -
rx(7,j)'*H(1,j) - rx(8,j)'*H(2,j);
            S4 = S4 + rx(1,j)*H(4,j)' - rx(2,j)*H(3,j)' +
rx(3,j)*H(2,j)' - rx(4,j)*H(1,j)' ...
                    +rx(5,j)'*H(4,j) - rx(6,j)'*H(3,j) +
rx(7,j)'*H(2,j) - rx(8,j)'*H(1,j);
            Hnorm = Hnorm + H(:,j)'*H(:,j);
        end
        for L=1:Mary
            Con = (-1 + 2*Hnorm) * Smap(L) * Smap(L)';
            est_s1(L)= real( (S1 - Smap(L))*(S1 - Smap(L))' +
Con);
            est_s2(L)= real( (S2 - Smap(L))*(S2 - Smap(L))' +
Con);
            est_s3(L)= real( (S3 - Smap(L))*(S3 - Smap(L))' +
Con);
            est_s4(L)= real( (S4 - Smap(L))*(S4 - Smap(L))' +
Con);
        end
        [A,B]=min(est_s1);
        shat(k)=B-1;
        [A,B]=min(est_s2);
        shat(k+1)=B-1;
        [A,B]=min(est_s3);
        shat(k+2)=B-1;
        [A,B]=min(est_s4);
        shat(k+3)=B-1;
    end
    err(index)=sum(shat~=symb');
```

Lampiran A5 Funtion Matrik H3

```
function [err]=MatrikH3(index,snr,nsymb,Mary,modtype,nRx,err);
    directory_utama;
    cd(dir_utama)
        Global_parameter
    cd(folder_PU)

    nTx=3;
    Set=[0:Mary-1]';
    Smap=dmodce(Set,1,1,modtype,Mary);
    symb=randsrc(nsymb,1,[0:Mary-1]);
    msg=dmodce(symb,1,1,modtype,Mary);
    Eav=Smap'*Smap/Mary;
    NF=10^(snr/10);
    S=sqrt(nTx*Eav/(2*NF));
    for k=1:3:nsymb,
        Sv=msg(k:k+2);
        F1=Sv(3)/sqrt(2);
        F2=(-Sv(1) - Sv(1)' + Sv(2) - Sv(2')) / 2;
        F3=(+Sv(2) + Sv(2)' + Sv(1) - Sv(1')) / 2;
        H3 = [+Sv(1)   Sv(2)   F1;
              -Sv(2)' Sv(1)' F1;
               F1'     F1'     F2;
               F1'    -F1'     F3];

        H=(randn(nTx,1) + i*randn(nTx,1))/sqrt(2);

        if check_OFDM == 1
            %OFDM
            input_OFDM1 =  H3*H;
            [v w] = size(input_OFDM1);
            panjang_sinyal_in = v;
            for y = 1:w
                input_OFDM = input_OFDM1(:,y);
                cd(folder_transceiver)
                    transceiver
                    noise = S*randn(1,size_fft);
                    sinyal_in_receiver = sinyal_out_transceiver +
noise;
                cd(folder_receiver)
                    receiver
                    rx(:,y) = sinyal_output_receiver;
                cd(folder_PU)
            end
        else
            noise=S*(randn(4,1) + i*randn(4,1));
            rx= H3*H + noise;
        end

        S1 = + rx(1)*H(1)' + rx(2)'*H(2) ...
             + (rx(4) - rx(3))*H(3)'/2 ...
             - (rx(3) + rx(4))'*H(3)/2;



S2 = + rx(1)*H(2)' - rx(2)'*H(1) ...
```

```
            + (rx(4) + rx(3))*H(3)'/2 ...
            + (-rx(3) + rx(4))'*H(3)/2;

        S3 = + (rx(1) + rx(2))*H(3)'/sqrt(2) ...
            + rx(3)'*(H(1) + H(2))/sqrt(2) ...
            + rx(4)'*(H(1) - H(2))/sqrt(2);


        for L=1:Mary
            Con = (-1 + H'*H) * Smap(L) * Smap(L)';
            est_s1(L)= real( (S1 - Smap(L))*(S1 - Smap(L))' +
Con);
            est_s2(L)= real( (S2 - Smap(L))*(S2 - Smap(L))' +
Con);
            est_s3(L)= real( (S3 - Smap(L))*(S3 - Smap(L))' +
Con);
        end
        [A,B]=min(est_s1);
        shat(k)=B-1;
        [A,B]=min(est_s2);
        shat(k+1)=B-1;
        [A,B]=min(est_s3);
        shat(k+2)=B-1;
    end
    err(index)=sum(shat~=symb');
```

Lampiran A6 Funtion Matrik H4

```
Pfunction [err]=MatrikH4(index,snr,nsymb,Mary,modtype,nRx,err);
    directory_utama;
    cd(dir_utama)
        Global_parameter
    cd(folder_PU)

    nTx=4;
    Set=[0:Mary-1]';
    Smap=dmodce(Set,1,1,'qask',Mary);
    symb=randsrc(nsymb,1,[0:Mary-1]);
    msg=dmodce(symb,1,1,'qask',Mary);
    Eav=Smap'*Smap/Mary;
    NF=10^(snr/10);
    S=sqrt(nTx*Eav/(2*NF));
    for k=1:3:nsymb,
        Sv=msg(k:k+2);
        F1  =  Sv(3)/sqrt(2);
        F33 = (-Sv(1) - Sv(1)' + Sv(2) - Sv(2)')/2;
        F34 = (-Sv(2) - Sv(2)' + Sv(1) - Sv(1)')/2;
        F43 = (+Sv(2) + Sv(2)' + Sv(1) - Sv(1)')/2;
        F44 = -(+Sv(1) + Sv(1)' + Sv(2) - Sv(2)')/2;

        H4 = [+Sv(1)   Sv(2)   F1    F1;
              -Sv(2)' Sv(1)' F1   -F1;
               F1'     F1'     F33   F34;
               F1'    -F1'     F43   F44];

        H=(randn(nTx,1) + i*randn(nTx,1))/sqrt(2);

        if check_OFDM == 1
            %OFDM
            input_OFDM1 =  H4*H;
            [v w] = size(input_OFDM1);
            panjang_sinyal_in = v;
            for y = 1:w
                input_OFDM = input_OFDM1(:,y);
                cd(folder_transceiver)
                    transceiver
                    noise = S*randn(1,size_fft);
                    sinyal_in_receiver = sinyal_out_transceiver +
noise;
                cd(folder_receiver)
                    receiver
                    rx(:,y) = sinyal_output_receiver;
                cd(folder_PU)
            end
        else
            noise=S*(randn(4,1) + i*randn(4,1));
            rx= H4*H + noise;
        end




S1 = + rx(1)*H(1)' + rx(2)'*H(2) ...
```

```
            + (rx(4) - rx(3)) * (H(3)' - H(4)') / 2 ...
            - (rx(3) + rx(4))' * (H(3) + H(4)) / 2;

        S2 = + rx(1)*H(2)' - rx(2)'*H(1) ...
            + (rx(4) + rx(3)) * (H(3)' - H(4)') / 2 ...
            + (-rx(3) + rx(4))' * (H(3) + H(4)) / 2;

        S3 = + (rx(1) + rx(2))*H(3)'/sqrt(2) + (rx(1) -
rx(2))*H(4)'/sqrt(2) ...
            + rx(3)'*(H(1) + H(2))/sqrt(2) ...
            + rx(4)'*(H(1) - H(2))/sqrt(2);


        for L=1:Mary
            Con = (-1 + H'*H) * Smap(L) * Smap(L)';
            est_s1(L)= real( (S1 - Smap(L))*(S1 - Smap(L))' +
Con);
            est_s2(L)= real( (S2 - Smap(L))*(S2 - Smap(L))' +
Con);
            est_s3(L)= real( (S3 - Smap(L))*(S3 - Smap(L))' +
Con);
        end
        [A,B]=min(est_s1);
        shat(k)=B-1;
        [A,B]=min(est_s2);
        shat(k+1)=B-1;
        [A,B]=min(est_s3);
        shat(k+2)=B-1;
    end
    err(index)=sum(shat~=symb');
```

Lampiran A7 Program Transceiver OFDM

```
function [B] = transceiver
    directory_utama;
    cd(dir_utama2)
        Global_parameter
    cd(folder_transceiver)
        jumlah_carrier = jml_carrier_OFDM;
        size_fft = ukuran_fft_OFDM;

        sinyal_in_transceiver         = input_OFDM;
        sinyal_in_transceiver_polar   =
sinyal_in_transceiver;%konversi_biner_ke_polar(sinyal_in_transceiv
er );

        %Start Transceiver
            transceiver_block_code
                %sinyal_out_transceiver_block_code
            %IFFT-----------------------------------------------
-----------------
                td_sets = zeros(jumlah_block_code,size_fft);
                for i = 1:jumlah_block_code
                    td_sets(i,1:size_fft) =
((ifft(sinyal_out_transceiver_block_code(i,1:size_fft))));
                end
                sinyal_output_IFFT = td_sets;
            %-----------------------------------------------
-----------------
            transceiver_block_decode;
        %End Transceiver
        sinyal_out_transceiver = xmit;
```

Lampiran A8 Program Receiver OFDM

```
function [F] = receiver
    directory_utama_rx;
    cd(dir_utama2)
        Global_parameter
    cd(folder_receiver)

    sinyal_input_receiver = sinyal_in_receiver;
    receiver_block_code
    %sinyal_out_receiver_block_code
    %FFT-------------------------------------------------------------
---------
        recv_spaced_block_code =
zeros(jumlah_block_code,size_fft);
        for i = 1:jumlah_block_code
            recv_spaced_block_code(i,1:size_fft) =
fft(sinyal_out_receiver_block_code(i,1:size_fft));
        end
        recv_spaced_block_code;
        sinyal_out_FFT =
recv_spaced_block_code;%receiver_spaced_block_code;
    %-------------------------------------------------------------
---------
    %-
    receiver_block_encode;
    %sinyal_output_receiver =
sinyal_output_receiver_block_encode;%konversi_polar_ke_biner(sinya
l_output_receiver_block_encode);
    sinyal_output_receiver =
reshape(recv_block_code,panjang_sinyal_in,1);
```