

**Lampiran A**  
**Data Masukan dan Data Target**









































**Lampiran B**  
**Isi Program dan Hasil Pelatihan**

## ❖ Pelatihan

```
clear;
close all;
clc;

% -----%
%% Program untuk klasifikasi sinyal radar
% -----%

% -----%
%% Sinyal masukan %
% -----%
for j=1:3
    [nama_file_masukan,path_file_masukan]=uigetfile('*.mat','Pilih file masukan');
    load (nama_file_masukan);
    x=f;
    x=abs(x);
    [brs kol]=size(x);
    for k=1:brs
        for m=1:kol
            if x(k,m)>0
                x(k,m)=1;
            else
                x(k,m)=-1;
            end;
        end;
    end;
    if j==1
        x1=x;
    elseif j==2
        x2=x;
    elseif j==3
        x3=x;
    end;
    clear x f;
end

clear j k m

masukan_1=x1(1:180,1);
masukan_2=x2(1:180,1);
masukan_3=x3(1:180,1);
bias=ones(1,180);

% -----%
%% Sinyal target %
% -----%
[nama_file_target,path_file_target]=uigetfile('*.mat','Pilih file target');
load (nama_file_masukan);
x=f;
x=abs(x);
[brs kol]=size(x);
for k=1:brs
    for m=1:kol
        if x(k,m)>0
```

```

        x(k,m)=1;
    else
        x(k,m)=-1;
    end;
end;
end;
x4=x;
target=x4(1:180,1)';
clear x f;

%-----%
% % Inisialisasi          %
%-----%
alpha=0.1;
bobot_bias=zeros(1,180);
bobot_masukan_1=zeros(1,180);
bobot_masukan_2=zeros(1,180);
bobot_masukan_3=zeros(1,180);
keluaran=zeros(1,180);

iterasi=0;

delta_bobot_masukan_1(1)=alpha.*(target(1)-keluaran(1)).*masukan_1(1);
delta_bobot_masukan_2(1)=alpha.*(target(1)-keluaran(1)).*masukan_2(1);
delta_bobot_masukan_3(1)=alpha.*(target(1)-keluaran(1)).*masukan_3(1);
delta_bobot_bias(1)=alpha.*(target(1)-keluaran(1));
maks=[delta_bobot_masukan_1(1),delta_bobot_masukan_2(1),bobot_masukan_3(1),...
    delta_bobot_bias(1)];
maks=max(maks);
disp(['Mulai iterasi']);

% temp=zeros(2,40);
% temp(1,1:40)=1:1:40;
while maks~=0
    disp(' ');
    iterasi=iterasi+1;
    disp(['Ini adalah epoch ke-',num2str(iterasi)]);
    net(1)=masukan_1(1).*bobot_masukan_1(1)+masukan_2(1).*bobot_masukan_2(1)+...
        masukan_3(1).*bobot_masukan_3(1)+bobot_bias(1);
    keluaran(1)=net(1);
    if keluaran(1)~=target(1)
        delta_bobot_masukan_1(1)=alpha.*(target(1)-keluaran(1)).*masukan_1(1);
        delta_bobot_masukan_2(1)=alpha.*(target(1)-keluaran(1)).*masukan_2(1);
        delta_bobot_masukan_3(1)=alpha.*(target(1)-keluaran(1)).*masukan_3(1);
        delta_bobot_bias(1)=alpha.*(target(1)-keluaran(1));
    else
        delta_bobot_masukan_1(1)=0;
        delta_bobot_masukan_2(1)=0;
        delta_bobot_masukan_3(1)=0;
        delta_bobot_bias(1)=0;
    end;
    bobot_masukan_1(1)=bobot_masukan_1(1)+delta_bobot_masukan_1(1);
    bobot_masukan_2(1)=bobot_masukan_2(1)+delta_bobot_masukan_2(1);
    bobot_masukan_3(1)=bobot_masukan_3(1)+delta_bobot_masukan_3(1);
    bobot_bias(1)=bobot_bias(1)+delta_bobot_bias(1);

    for j=2:brs,j;

```

```

net(j)=masukan_1(j).*bobot_masukan_1(j-1)+masukan_2(j).*bobot_masukan_2(j-1)+...
    masukan_3(j).*bobot_masukan_3(j-1)+bobot_bias(j-1);
keluaran(j)=net(j);
if keluaran(j)~=target(j)
    delta_bobot_masukan_1(j)=alpha.*(target(j)-keluaran(j)).*masukan_1(j);
    delta_bobot_masukan_2(j)=alpha.*(target(j)-keluaran(j)).*masukan_2(j);
    delta_bobot_masukan_3(j)=alpha.*(target(j)-keluaran(j)).*masukan_3(j);
    delta_bobot_bias(j)=alpha.*(target(j)-keluaran(j));
else
    delta_bobot_masukan_1(j)=0;
    delta_bobot_masukan_2(j)=0;
    delta_bobot_masukan_3(j)=0;
    delta_bobot_bias(j)=0;
end;
bobot_masukan_1(j)=bobot_masukan_1(j-1)+delta_bobot_masukan_1(j);
bobot_masukan_2(j)=bobot_masukan_2(j-1)+delta_bobot_masukan_2(j);
bobot_masukan_3(j)=bobot_masukan_3(j-1)+delta_bobot_masukan_3(j);
bobot_bias(j)=bobot_bias(j-1)+delta_bobot_bias(j);
end;
bobot_masukan_1(1)=bobot_masukan_1(brs);
bobot_masukan_2(1)=bobot_masukan_2(brs);
bobot_masukan_3(1)=bobot_masukan_3(brs);
bobot_bias(1)=bobot_bias(brs);
maks=[delta_bobot_masukan_1(brs),delta_bobot_masukan_2(brs),...
    delta_bobot_masukan_3(brs),delta_bobot_bias(brs)];
maks=max(maks);
maks_skala=maks.*1e12
maks_plot(iterasi)=maks_skala;
temp(1,iterasi)=iterasi;
temp(2,iterasi)=maks_skala;
disp(' ');
j;
clear j;
%     pause;

end;

end;

%-----%
%% Gambar kurva kesalahan terhadap epoch
%-----%
figure;
plot(1:iterasi,maks_plot(1:iterasi));grid;
title('Hasil simulasi pertama');
xlabel('Epoch ke-');ylabel('Kesalahan');
x_maks=iterasi+2;
y_maks=max(maks_plot)+5e-14;
axis([0 x_maks -5e-13 y_maks])

```