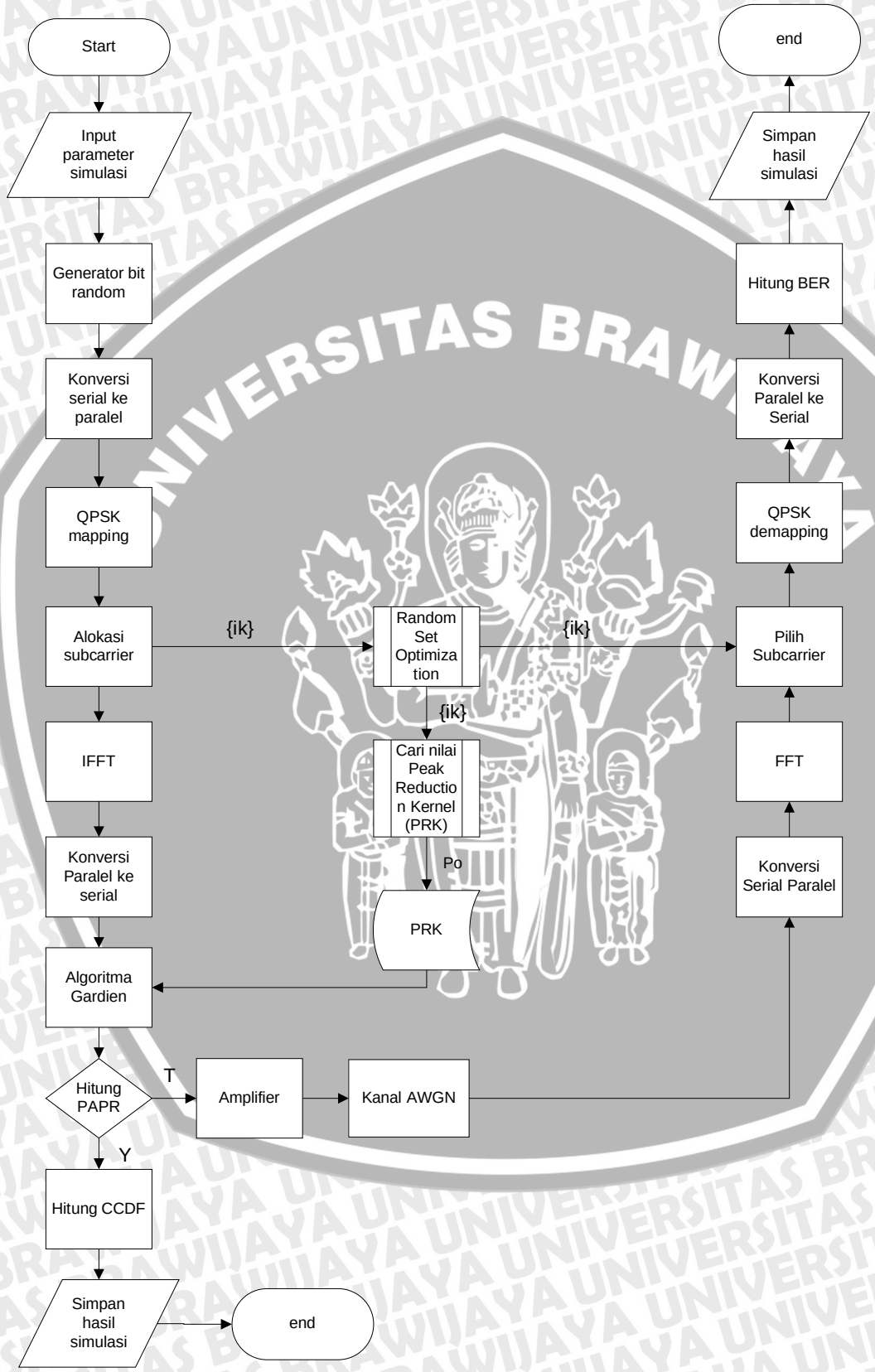
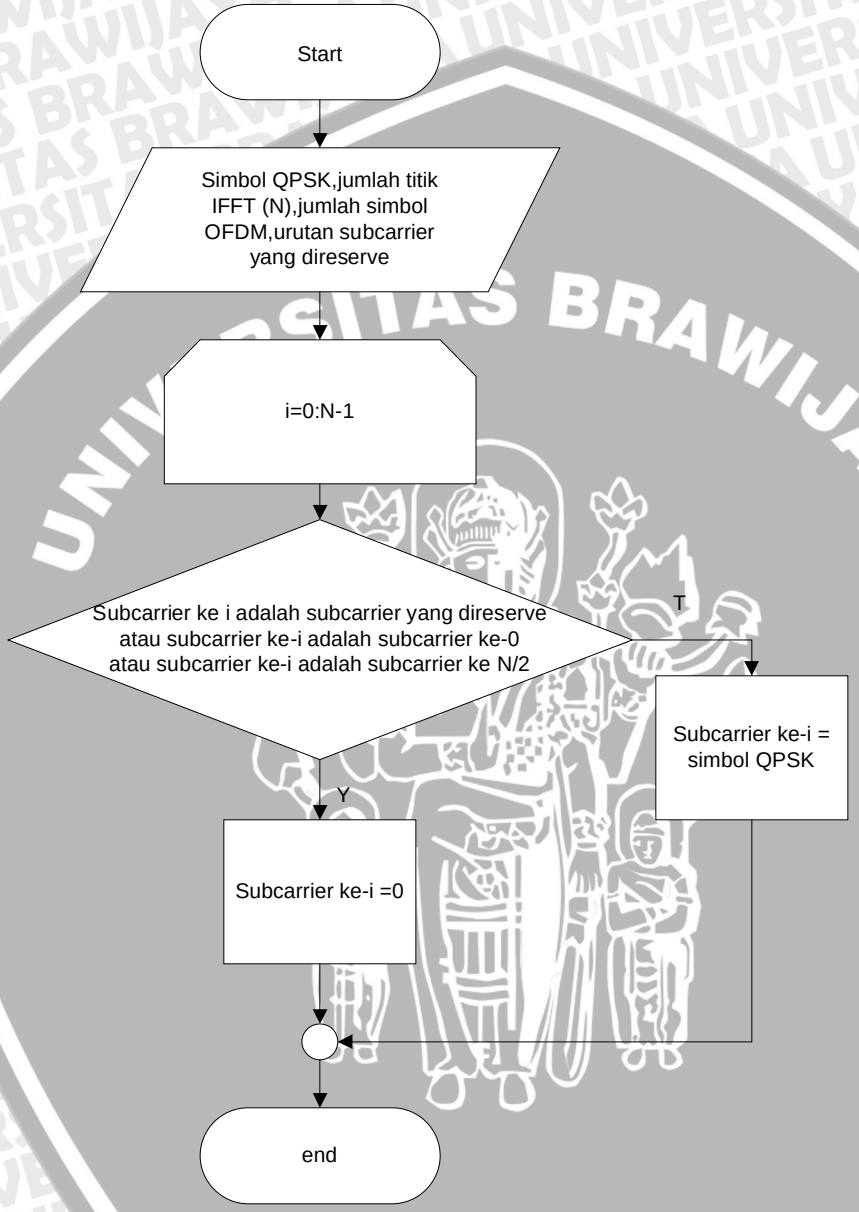
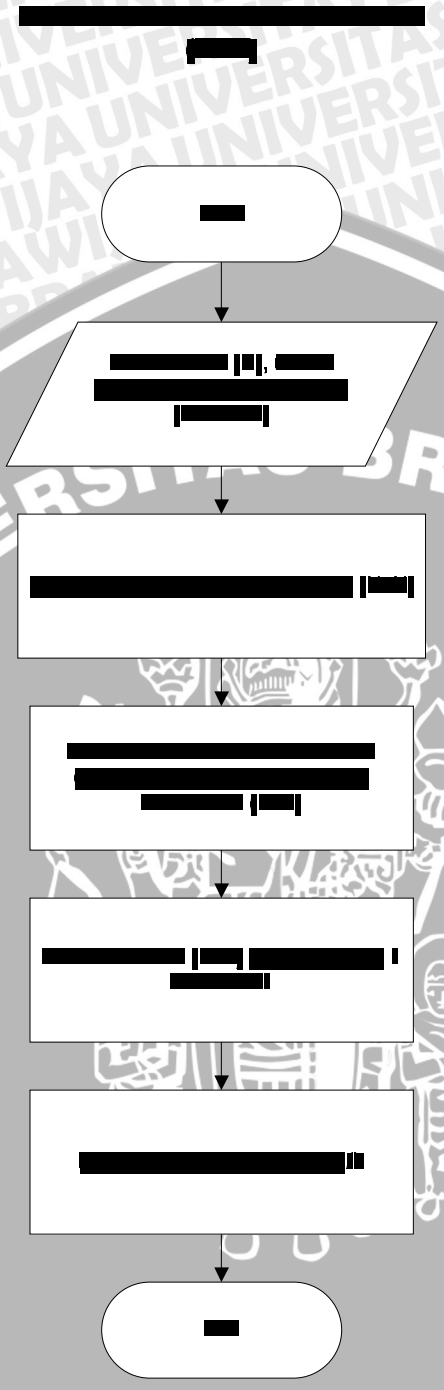
Simulasi Tone Reservation pada OFDM



Flow Chart Alokasi Subcarrier





```
%Script for computing the per symbol peak to average PAPR for
% an OFDM transmit waveform (loosely based on IEEE 802.11A
% specifications)
% Further, Cumulative Distribution Function (CDF) plots of the
% PAPR is captured

clear
nFFTSize = 64;
% for each symbol bits a1 to a52 are assigned to subcarrier
% index [-26 to -1 1 to 26]
subcarrierIndex = [-26:-1 1:26];
nBit = 10000;
ip = rand(1,nBit) > 0.5; % generating 1's and 0's
nBitPerSymbol = 52;

nSymbol = ceil(nBit/nBitPerSymbol);

% BPSK modulation
% bit0 --> -1
% bit1 --> +1
ipMod = 2*ip - 1;
ipMod = [ipMod zeros(1,nBitPerSymbol*nSymbol-nBit)];
ipMod = reshape(ipMod,nSymbol,nBitPerSymbol);

st = []; % empty vector

for ii = 1:nSymbol

inputiFFT = zeros(1,nFFTSize);

% assigning bits a1 to a52 to subcarriers [-26 to -1, 1 to 26]
inputiFFT(subcarrierIndex+nFFTSize/2+1) = ipMod(ii,:);

%  shift subcarriers at indices [-26 to -1] to fft input indices [38 to
63]
inputiFFT = fftshift(inputiFFT);

outputiFFT = 64*ifft(inputiFFT,nFFTSize);

% adding cyclic prefix of 16 samples
outputiFFT_with_CP = [outputiFFT(49:64) outputiFFT];

% computing the peak to average power ratio for each symbol
meanSquareValue = outputiFFT*outputiFFT'/length(outputiFFT);
peakValue = max(outputiFFT.*conj(outputiFFT));
paprSymbol(ii) = peakValue/meanSquareValue;

% concatenating the symbols to form the final output
st = [st outputiFFT_with_CP];

end

close all
paprSymboldB = 10*log10(paprSymbol);
```

```
[n x] = hist(paprSymboldB,[0:0.5:15]);
plot(x,cumsum(n)/nSymbol,'LineWidth',4)
xlabel('papr, x dB')
ylabel('Probability, X <=x')
title('CDF plots of PAPR from an IEEE 802.11a Tx with BPSK modulation')
grid on

fungsi trtonedealoc()
function[iout,qout]=trtonedealoc(idata,qdata,fftlen,L,nd,ik)

%******************variabel*************
%idata
%qdata
%iout
%qout
%fftlen
%L
%nd
%ik
%*******************************************************

idata1=zeros((ffteln-L-2),nd);
qdata1=zeros((ffteln-L-2),nd);

n=1;

for m=2:fftlen/2
if (isempty(find(ik==m)))
idata1(n,:)=idata(m,:);
qdata1(n,:)=qdata(m,:);
n=n+1;
end
end

for m=2+fftlen/2:fftlen
if (isempty(find(ik==m)))
idata1(n,:)=idata(m,:);
qdata1(n,:)=qdata(m,:);
n=n+1;
end
end

iout=idata1;
qout=qdata1;
%************************end of file**********************
fungsi qpskdemod ()
function[demodata]=qpskdemod(idata,qdata,para,nd,m1)
%******************variabel*******************
%idata
%qdata
%demodata)
%para
%nd
%m1
%(QPSK-2 16 QAM-4)
%*************************************************
```

```
demodata=zeros(para,m1*nd);
demodata((1:para),(1:m1:m1*nd-1))=idata((1:para),(1:nd))>=0;
demodata((1:para),(2:m1:m1*nd))=qdata((1:para),(1:nd))>=0
%********************end of file******************
```