```
%Explanation of base program

%The program splits a squid into two junctions that have xmax discrete sections.  Each
%section has a supercurrent density, and a phase difference that is
%contributed to by the field (PhaseF) in the junction, the field in the squid loop,
%and an arbitrary phase set at x=1 %called Phase1.  These phase factors are
%all summed along the junction to determine the local phase difference
%across the junction at each point.  For each value of field and Phase1 the
%supercurrent at each point is calculated and the net supercurrent (sum
%along the junction and then the sum of the two junctions together)
%is calcualted to find the supercurrent carried at that phase and that
%field.  To find the maximum supercurrent for a given field, the max of
%that vector is taken for that field value.  The minimum can be calculated
%to find the negative critical current.  Both of these can be plotted
%against the magnetic field that is applied.  This is the exact measurment
%of the critical current vs field that we do in the lab.

%This version is the most basic version of the squid. It runs through
%Junction asymmetry.

% Critical Current Asymmetry
% Junction Area Asymmetry
% Critical Current density variation along each junction
% Squid Loop size variation
% Changes in the field scan range



%Abreviations used
%Junction=Junc
%Super Current = SCur or just SC
%Step Size = SS sufix
%Width = Wid
%Length = Len
%Magnitude = Mag


%% Clearing memory and input screen

clear;
clc;
close all;

%% Defining the Parameters of the Simulaiton

    %Dividing the Junctions up into discrete sections
        xmax1=51;
        xmax2=51;
        x1(1,:)=(1:xmax1);
        x2(1,:)=(1:xmax2);
```

```
    %Critical Current Magnitudes
        SCurrentMag1=1;
        SCurrentMag2=1;

        SCurNoiseMag1=.01;
        SCurNoiseMag2=.01;

    %Setting Squid Loop Parameers
        LoopWid=1;
        LoopLen=5;
    %Junction Area Dimensions
        JuncWid1=.01;
        JuncLen1=.01;

        JuncWid2=.01;
        JuncLen2=.01;

%Setting up Loop Parameters

    %Phase Loop parameters
        p=1;
        pmax=101;
        Phase0Min=0*pi;
        Phase0Max=2*pi;

    %Field Parameters
        f=1;
        fmax=1001;
        FieldMin=-2;
        FieldMax=2;

%Calculating Critical Current Densities
    JuncArea1=JuncWid1*JuncLen1;
    JuncArea2=JuncWid2*JuncLen2;
    LoopArea=LoopWid*LoopLen;

    SCurNoise1=SCurNoiseMag1*(2*rand(1,xmax1)-1);
    SCurNoise2=SCurNoiseMag2*(2*rand(1,xmax2)-1);

    SCurDen1=SCurrentMag1*ones(1,xmax1)/xmax1+SCurNoise1/xmax1;
    SCurDen2=SCurrentMag2*ones(1,xmax2)/xmax2+SCurNoise2/xmax2;

%Pre Allocating memory to the arrays (should decrease runtime)
    Phase0=zeros(1,pmax);
    Field=zeros(1,fmax);
    FluxinJunc1=zeros(1,fmax);
    FluxinJunc2=zeros(1,fmax);
    FluxinLoop=zeros(1,fmax);

    PhaseFDen1=zeros(1,fmax);
    PhaseFDen2=zeros(1,fmax);
```

```
    PhaseFL=zeros(1,fmax);

    SCurrent1=zeros(xmax1,pmax,fmax);
    SCurrent2=zeros(xmax2,pmax,fmax);
    SCurrentNet=zeros(1,pmax);

    MaxSCurrentNet=zeros(1,fmax);
    MinSCurrentNet=zeros(1,fmax);
%% Loops for running the simulation Meat of the Simulation


    %Field Contribution to the Phase
    %Define the Field ForLoop setp size, then run the Field for ForLoop
    FieldSS=(FieldMax-FieldMin)/(fmax-1);
    for f=1:fmax

        Field(f)=FieldMin+(f-1)*FieldSS;


        PhaseF1=2*pi*Field(f)*JuncArea1;
        PhaseF2=2*pi*Field(f)*JuncArea2;
        PhaseFL=2*pi*Field(f)*LoopArea;

        PhaseFDen1=PhaseF1*x1/xmax1;
        PhaseFDen2=PhaseF2*x2/xmax2;

        %Phase0 ForLoop of externally set phase
        %Define the Phase0 setp size, then run the ForLoop
        Phase0SS=(Phase0Max-Phase0Min)/(pmax-1);
        for p=1:pmax

            Phase0(p)=Phase0Min+(p-1)*Phase0SS;

            PhaseDrop1=Phase0(p)+PhaseFDen1;
            PhaseDrop2=Phase0(p)+PhaseF1+PhaseFL+PhaseFDen2;

            SCurrent1=SCurDen1.*sin(PhaseDrop1);
            SCurrent2=SCurDen2.*sin(PhaseDrop2);

            SCurrentNet(p)=sum(SCurrent1)+sum(SCurrent2);


        end
        MaxSCurrentNet(f)=max(SCurrentNet);
        MinSCurrentNet(f)=min(SCurrentNet);
    end



hold on
plot(Field,MaxSCurrentNet,'.')
```

```
xlabel('Magnetic Field'); ylabel('Critical Current');

hold on
plot(Field,MinSCurrentNet,'.')
```