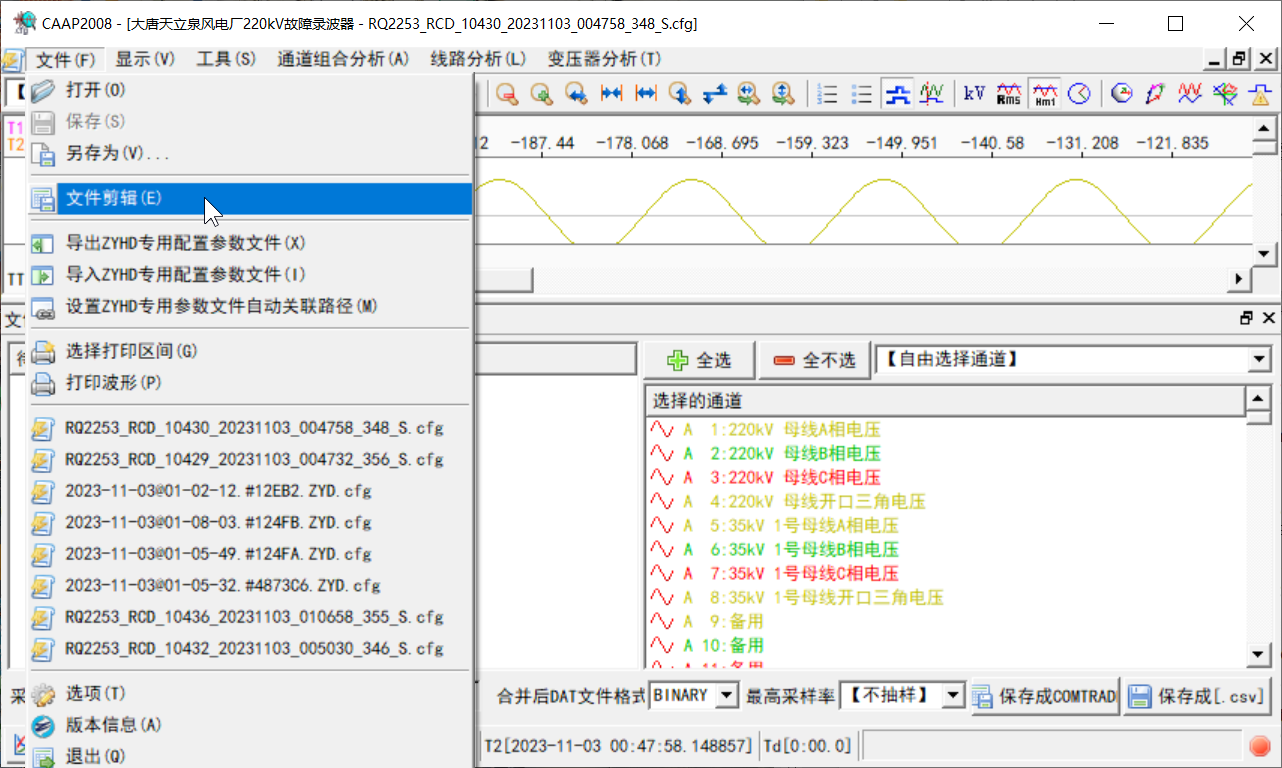
% 首先需要在故录查看的软件里把1sec长度的数据导成csv



%% verion 2.1 revised by Yamin

% This version, read csv as input, you have to specify the column of Uabc

% and Iabc

close all;clc

%% User define zone

%MatUrl就是导出的csv的路径

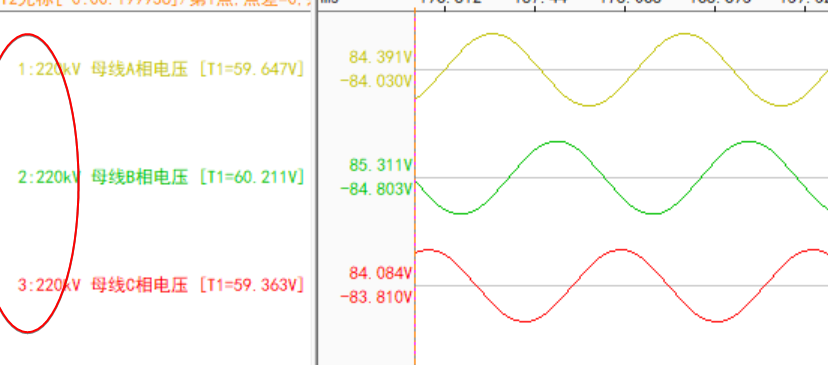
MatUrl = 'D:\Travail\RE\HIL\[Routine] MMC-HVDC\20230411\_ZhongDu\02-Data\Recorder\jishancun\2023-04-11@09-48-36.369500.csv';

% Define which column you select as Uabc and Iabc

% 这里定义电压、电流所在的列数，软件里左侧这个序号就是，比如这个就填USabcIdx = 1:3，电流也是一样

USabcIdx = 3:5;

ISabcIdx = 28:30;



% which freq do you want to focus on

% 这里填需要计算哪些频率，可以多写几个，长度可变

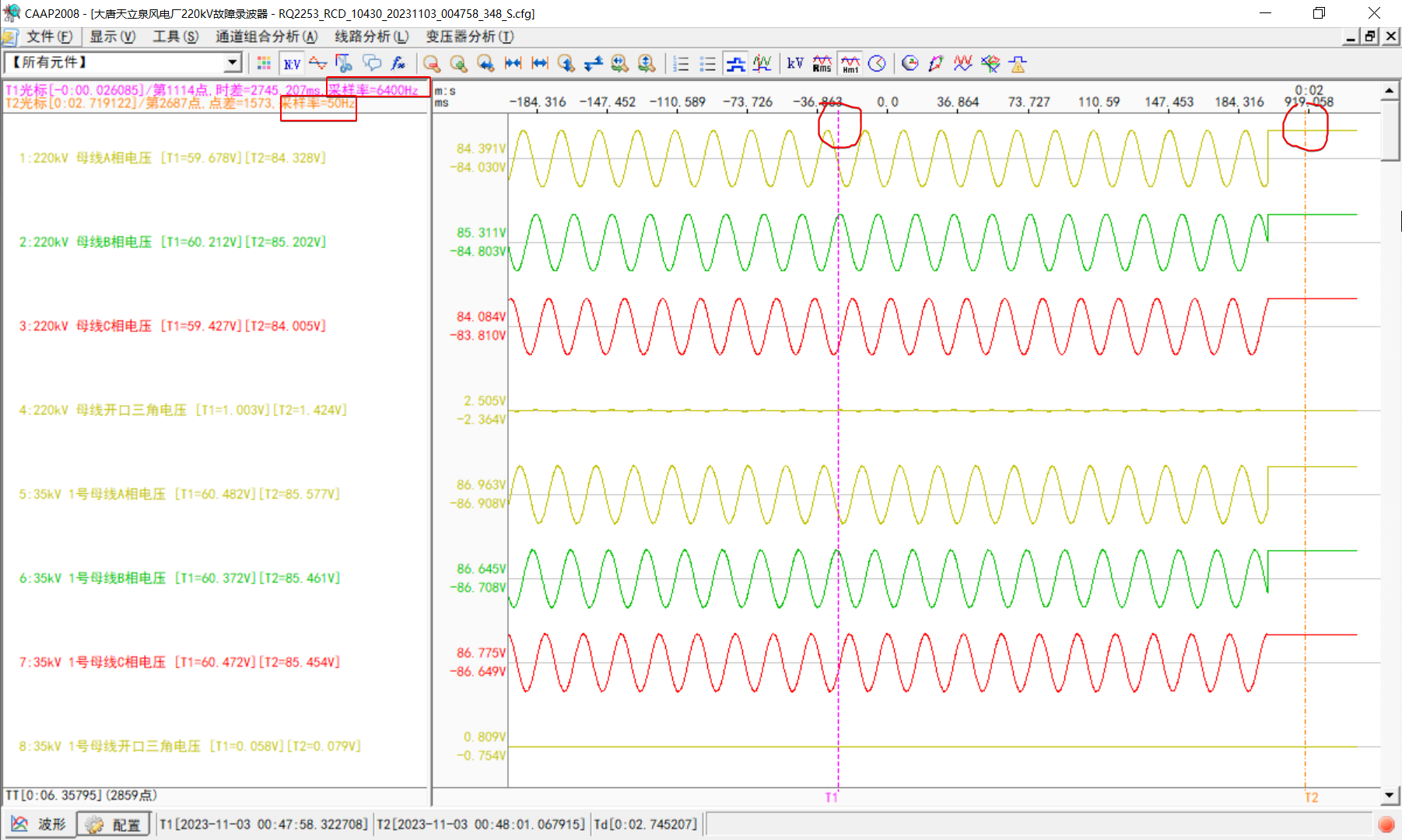
WHICH\_FREQ = [43, 50, 57];

% Sampling frequency, must consist with CSV

% SAMPLE\_FREQ，填故录数据的采样频率，这里要看你选的故录哪一段，故录采样率是会变的，不要让这1sec的csv数据包含不同的采样率

SAMPLE\_FREQ = 2e3;

T\_SPAN = 1; % signal real time span for selected window



RawData = table2array(readtable(MatUrl));

USabc = RawData(:, USabcIdx);

ISabc = RawData(:, ISabcIdx);

%% Don't touch the following code unless you know what are you doing

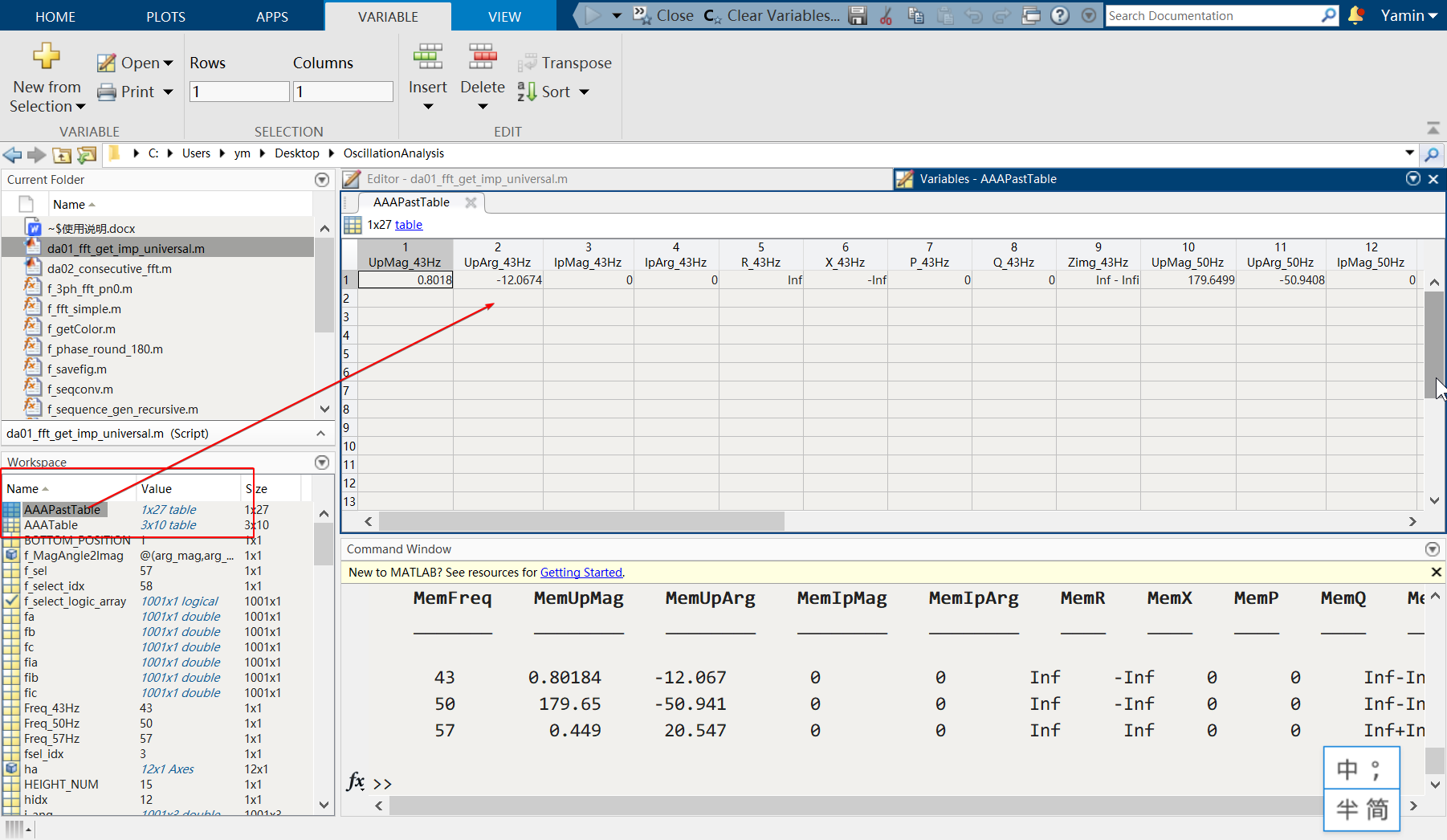
f\_MagAngle2Imag = @(arg\_mag, arg\_angle) arg\_mag .\* exp(1i .\* arg\_angle);

% 1: Voltage only, 2: Current only, 3: Impedance computing

% 工作模式：1：只计算电压FFT；2：只计算电流FFT；3：计算阻抗和功率

WORKING\_MODE = 3;

% 剩下的就没什么了，运行完之后会出图，具体数据看在这两个变量里。具体Mag是幅值，Arg是相角，依此类推



%% Your data input here

uabc\_iabc = [USabc, ISabc];

RAW\_DATA = uabc\_iabc;

SAMPLE\_TIME = 1/SAMPLE\_FREQ;

[RAW\_DATA\_RNum, RAW\_DATA\_CNum] = size(RAW\_DATA);

assert(RAW\_DATA\_CNum == 6, 'Please make sure that your signal is n-by-6 matrix')

%% generate fake time array

t\_array = 0:SAMPLE\_TIME:(RAW\_DATA\_RNum-1)\*SAMPLE\_TIME;

t\_array = t\_array';

uabc = RAW\_DATA(:, 1:3);

iabc = RAW\_DATA(:, 4:6);

L = T\_SPAN \* SAMPLE\_FREQ; % lenght of signal

N = L; % number of signal

scan = [t\_array, uabc, iabc];

Uas = uabc(:, 1);

Ubs = uabc(:, 2);

Ucs = uabc(:, 3);

Ias = iabc(:, 1);

Ibs = iabc(:, 2);

Ics = iabc(:, 3);

[fa, umag\_a, uang\_a] = f\_fft\_simple(Uas, SAMPLE\_FREQ, N);

[fb, umag\_b, uang\_b] = f\_fft\_simple(Ubs, SAMPLE\_FREQ, N);

[fc, umag\_c, uang\_c] = f\_fft\_simple(Ucs, SAMPLE\_FREQ, N);

[fia, imag\_a, iang\_a] = f\_fft\_simple(Ias, SAMPLE\_FREQ, N);

[fib, imag\_b, iang\_b] = f\_fft\_simple(Ibs, SAMPLE\_FREQ, N);

[fic, imag\_c, iang\_c] = f\_fft\_simple(Ics, SAMPLE\_FREQ, N);

u\_mag = [umag\_a, umag\_b, umag\_c];

% u\_ang = [uang\_a, uang\_b, uang\_c];

u\_ang = [uang\_a, uang\_b, uang\_c] + pi/2;

i\_mag = [imag\_a, imag\_b, imag\_c];

% i\_ang = [iang\_a, iang\_b, iang\_c];

i\_ang = [iang\_a, iang\_b, iang\_c] + pi/2;

%% convert fft result from mag/angle to complex

u\_complex\_array = f\_MagAngle2Imag(u\_mag, u\_ang);

i\_complex\_array = f\_MagAngle2Imag(i\_mag, i\_ang);

%% convert fft result from ABC complex form to positive-negative-zero form

u\_pn0\_array = f\_seqconv(u\_complex\_array');

u\_pn0\_array = u\_pn0\_array';

i\_pn0\_array = f\_seqconv(i\_complex\_array');

i\_pn0\_array = i\_pn0\_array';

z\_pn0\_array = u\_pn0\_array ./ i\_pn0\_array;

u\_pn0\_mag = abs(u\_pn0\_array);

u\_pn0\_ang = angle(u\_pn0\_array);

i\_pn0\_mag = abs(i\_pn0\_array);

i\_pn0\_ang = angle(i\_pn0\_array);

z\_pn0\_mag = abs(z\_pn0\_array);

z\_pn0\_ang = angle(z\_pn0\_array);

MemFreq = [];

MemUpMag = [];

MemUpArg = [];

MemIpMag = [];

MemIpArg = [];

MemR = [];

MemX = [];

MemP = [];

MemQ = [];

MemZimg = [];

AAAPastTable = table();

for fsel\_idx = 1:length(WHICH\_FREQ)

%% Select specific freq value

f\_sel = WHICH\_FREQ(fsel\_idx);

fprintf('FFT抽取%6.2f (Hz)频率，原始数据采样频率%i (Hz), 时间长度为%6.2f (sec)\n', ...

f\_sel, SAMPLE\_FREQ, T\_SPAN)

f\_select\_logic\_array = fa == f\_sel;

f\_select\_idx = find(f\_select\_logic\_array);

u\_mag\_sel = u\_mag(f\_select\_idx, :);

u\_ang\_sel = u\_ang(f\_select\_idx, :);

i\_mag\_sel = i\_mag(f\_select\_idx, :);

i\_ang\_sel = i\_ang(f\_select\_idx, :);

z\_pn0\_sel = z\_pn0\_array(f\_select\_idx, :);

u\_pn0\_mag\_sel = u\_pn0\_mag(f\_select\_idx, :);

u\_pn0\_ang\_sel = u\_pn0\_ang(f\_select\_idx, :);

i\_pn0\_mag\_sel = i\_pn0\_mag(f\_select\_idx, :);

i\_pn0\_ang\_sel = i\_pn0\_ang(f\_select\_idx, :);

z\_pn0\_mag\_sel = abs(z\_pn0\_sel);

z\_pn0\_ang\_sel = angle(z\_pn0\_sel);

Pp = 1.5\*u\_pn0\_mag\_sel(1)\*i\_pn0\_mag\_sel(1)\*cos(u\_pn0\_ang\_sel(1) - i\_pn0\_ang\_sel(1));

Qp = 1.5\*u\_pn0\_mag\_sel(1)\*i\_pn0\_mag\_sel(1)\*sin(u\_pn0\_ang\_sel(1) - i\_pn0\_ang\_sel(1));

%% Data store zone

MemFreq = [MemFreq; f\_sel];

MemUpMag = [MemUpMag; u\_pn0\_mag\_sel(1)];

MemUpArg = [MemUpArg; u\_pn0\_ang\_sel(1)\*180/pi];

MemIpMag = [MemIpMag; i\_pn0\_mag\_sel(1)];

MemIpArg = [MemIpArg; i\_pn0\_ang\_sel(1)\*180/pi];

MemR = [MemR; real(z\_pn0\_sel(1))];

MemX = [MemX; imag(z\_pn0\_sel(1))];

MemP = [MemP; Pp];

MemQ = [MemQ; Qp];

MemZimg = [MemZimg; z\_pn0\_sel(1)];

%%

eval(strcat('Freq\_', num2str(f\_sel), 'Hz = f\_sel'))

eval(strcat('UpMag\_', num2str(f\_sel), 'Hz = u\_pn0\_mag\_sel(1);'))

eval(strcat('UpArg\_', num2str(f\_sel), 'Hz = u\_pn0\_ang\_sel(1)\*180/pi;'))

eval(strcat('IpMag\_', num2str(f\_sel), 'Hz = i\_pn0\_mag\_sel(1);'))

eval(strcat('IpArg\_', num2str(f\_sel), 'Hz = i\_pn0\_ang\_sel(1)\*180/pi;'))

eval(strcat('R\_', num2str(f\_sel), 'Hz = real(z\_pn0\_sel(1));'))

eval(strcat('X\_', num2str(f\_sel), 'Hz = imag(z\_pn0\_sel(1));'))

eval(strcat('P\_', num2str(f\_sel), 'Hz = Pp;'))

eval(strcat('Q\_', num2str(f\_sel), 'Hz = Qp;'))

eval(strcat('Zimg\_', num2str(f\_sel), 'Hz = z\_pn0\_sel(1);'))

% eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'Freq\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'UpMag\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'UpArg\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'IpMag\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'IpArg\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'R\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'X\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'P\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'Q\_', num2str(f\_sel), 'Hz);') )

eval(strcat("AAAPastTable = addvars(AAAPastTable, ", 'Zimg\_', num2str(f\_sel), 'Hz);') )

%% Print out zone

disp("----- Sequence Result: Positive Negative Zero ------")

fprintf("Up mag, angle(degree): %f, %f\n", u\_pn0\_mag\_sel(1), u\_pn0\_ang\_sel(1)\*180/pi)

fprintf("Ip mag, angle(degree): %f, %f\n", i\_pn0\_mag\_sel(1), i\_pn0\_ang\_sel(1)\*180/pi)

fprintf("P = %f, Q = %f\n", Pp, Qp)

fprintf("Zp complex: %s\n", num2str(z\_pn0\_sel(1)))

end

AAATable = table(MemFreq, MemUpMag, MemUpArg, MemIpMag, MemIpArg, ...

MemR, MemX, MemP, MemQ, MemZimg)

% disp("-------")

% fprintf("Zp mag, angle(degree): %f, %f\n", z\_pn0\_mag\_sel(1), z\_pn0\_ang\_sel(1)\*180/pi)

% fprintf("Un mag, angle(degree): %f, %f\n", u\_pn0\_mag\_sel(2), u\_pn0\_ang\_sel(2)\*180/pi)

% fprintf("In mag, angle(degree): %f, %f\n", i\_pn0\_mag\_sel(2), i\_pn0\_ang\_sel(2)\*180/pi)

% fprintf("Zn mag, angle(degree): %f, %f\n", z\_pn0\_sel(2), z\_pn0\_sel(2)\*180/pi)

% fprintf("Zn complex: %s\n", num2str(z\_pn0\_sel(2)))

%

% fprintf("U0 mag, angle(degree): %f, %f\n", u\_pn0\_mag\_sel(3), u\_pn0\_ang\_sel(3)\*180/pi)

% fprintf("I0 mag, angle(degree): %f, %f\n", i\_pn0\_mag\_sel(3), i\_pn0\_ang\_sel(3)\*180/pi)

% fprintf("Z0 mag, angle(degree): %f, %f\n", z\_pn0\_sel(3), z\_pn0\_sel(2)\*180/pi)

% fprintf("Z0 complex: %s\n", num2str(z\_pn0\_sel(3)))

% convert ABC to positive-negative-zero sequence

%% Figure plot zone

LEFT\_POSITION = 5;

BOTTOM\_POSITION = 1;

WIDTH\_NUM = 12;

HEIGHT\_NUM = 15;

POSITION\_SET = [LEFT\_POSITION, BOTTOM\_POSITION, WIDTH\_NUM, HEIGHT\_NUM];

% Plot Voltage

figure;

subplot(3, 1, 1)

plot(fa, umag\_a, 'LineWidth', 1.5)

xlabel('频率 (Hz)')

ylabel('U (kV)')

title('Ua')

grid on

set(gca, 'XLim', [0, 100])

subplot(3, 1, 2)

plot(fb, umag\_b, 'LineWidth', 1.5)

xlabel('频率 (Hz)')

ylabel('U (kV)')

title('Ub')

grid on

set(gca, 'XLim', [0, 100])

subplot(3, 1, 3)

plot(fc, umag\_c, 'LineWidth', 1.5)

xlabel('频率 (Hz)')

ylabel('U (kV)')

title('Uc')

grid on

set(gca, 'XLim', [0, 100])

set(gcf,'unit','centimeters','position', POSITION\_SET)

% Plot Current

figure

subplot(3, 1, 1)

plot(fia, imag\_a, 'LineWidth', 1.5)

xlabel('频率 (Hz)')

ylabel('I (kA)')

title('Ia')

grid on

set(gca, 'XLim', [0, 100])

subplot(3, 1, 2)

plot(fib, imag\_b, 'LineWidth', 1.5)

xlabel('频率 (Hz)')

ylabel('I (kA)')

title('Ib')

grid on

set(gca, 'XLim', [0, 100])

subplot(3, 1, 3)

plot(fic, imag\_c, 'LineWidth', 1.5)

xlabel('频率 (Hz)')

ylabel('I (kA)')

title('Ic')

grid on

set(gca, 'XLim', [0, 100])

set(gcf,'unit','centimeters','position', POSITION\_SET)

% figure;

%

% subplot(2, 6, 1)

% plot(fa, u\_pn0\_mag(:, 1))

% title('U Pos mag')

% subplot(2, 6, 2)

% plot(fa, u\_pn0\_ang(:, 1)\*180/pi)

% title('U Pos phase')

%

% subplot(2, 6, 3)

% plot(fa, u\_pn0\_mag(:, 2))

% title('U Neg mag')

% subplot(2, 6, 4)

% plot(fa, u\_pn0\_ang(:, 2)\*180/pi)

% title('U Neg phase')

%

% subplot(2, 6, 5)

% plot(fa, i\_pn0\_mag(:, 1))

% title('I Pos mag')

% subplot(2, 6, 6)

% plot(fa, i\_pn0\_ang(:, 1)\*180/pi)

% title('I Pos phase')

%

% subplot(2, 6, 7)

% plot(fa, i\_pn0\_mag(:, 2))

% title('I Neg mag')

% subplot(2, 6, 8)

% plot(fa, i\_pn0\_ang(:, 2)\*180/pi)

% title('I Neg phase')

%

% subplot(2, 6, 9)

% plot(fa, i\_pn0\_mag(:, 1))

% title('Z Pos mag')

% subplot(2, 6, 10)

% plot(fa, i\_pn0\_ang(:, 1)\*180/pi)

% title('Z Pos phase')

%

% subplot(2, 6, 11)

% plot(fa, i\_pn0\_mag(:, 2))

% title('Z Neg mag')

% subplot(2, 6, 12)

% plot(fa, i\_pn0\_ang(:, 2)\*180/pi)

% title('Z Neg phase')

% set(gcf, "Position", [20, 20, 18, 6])

XLIM = [0, 100];

SUBPLOT\_FONT\_SIZE = 10;

figure

[ha, pos] = tight\_subplot(4, 3, [0.06, 0.04], [0.03, 0.03], [0.04, 0.01]);

set(gcf,'unit','centimeters','position',[0.5, 1, 45, 20])

axes(ha(1))

title('test')

for hidx = 1:12

axes(ha(hidx))

switch hidx

%% positive

case 1

plot(fa, u\_pn0\_mag(:, 1))

grid on

title('U Pos Mag')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 4

plot(fa, u\_pn0\_ang(:, 1)\*180/pi)

grid on

title('U Pos Phase')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 7

plot(fa, i\_pn0\_mag(:, 1))

grid on

title('I Pos Mag')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 10

plot(fa, i\_pn0\_ang(:, 1)\*180/pi)

grid on

title('I Pos Phase')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

%% negative

case 2

plot(fa, u\_pn0\_mag(:, 2))

grid on

title('U Neg Mag')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 5

plot(fa, u\_pn0\_ang(:, 2)\*180/pi)

grid on

title('U Neg Phase')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 8

plot(fa, i\_pn0\_mag(:, 2))

grid on

title('I Neg Mag')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 11

plot(fa, i\_pn0\_ang(:, 2)\*180/pi)

grid on

title('I Neg Phase')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 3

plot(fa, z\_pn0\_mag(:, 1))

grid on

title('Z Pos Mag')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 6

plot(fa, z\_pn0\_ang(:, 1)\*180/pi)

grid on

title('Z Pos Phase')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 9

plot(fa, z\_pn0\_mag(:, 2))

grid on

title('Z Neg Mag')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

case 12

plot(fa, z\_pn0\_ang(:, 2)\*180/pi)

grid on

title('Z Neg Phase')

set(gca,'XLim', XLIM, 'fontname', 'times new roman', 'fontsize', SUBPLOT\_FONT\_SIZE)

otherwise

disp('wrong')

end

end