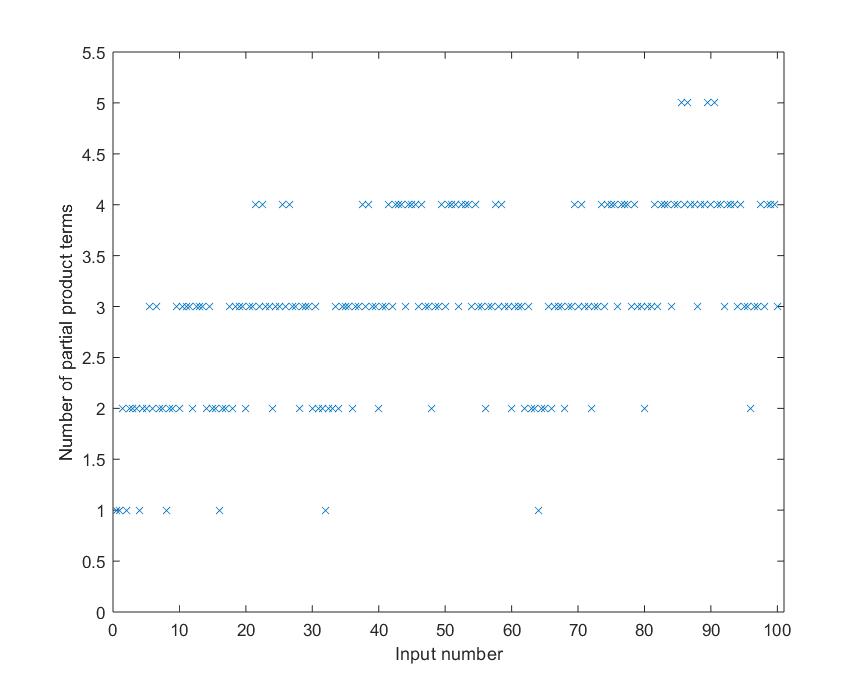
# Problem 1

## Result

Total sum for +0.5 - +100.00 = 605



## numppterms.m

function [mpp] = numppterms(num)

mpp = 0;

rem = num;

while (rem ~= 0 && mpp < 10)

for i = -1:10

if 2^i >= abs(rem)

mpp = mpp + 1;

if (2^i - abs(rem) < abs(rem) - 2^(i-1))

rem = abs(rem) - 2^i;

else

rem = abs(rem) - 2^(i-1);

end

break;

end

end

end

end

# Problem 2

## Matlab output for unscaled coefficients

Scale 1.000000 Using coefficients [ 17 -90 241 902 241 -90 17]

coef #1 17, 2 partial products

coef #2 -90, 4 partial products

coef #3 241, 3 partial products

coef #4 902, 4 partial products

coef #5 241, 3 partial products

coef #6 -90, 4 partial products

coef #7 17, 2 partial products

In total, need 22 partial products

## Matlab output for scaled coefficients

Scale 0.532000 Using coefficients [ 9 -48 128 480 128 -48 9]

coef #1 9, 2 partial products

coef #2 -48, 2 partial products

coef #3 128, 1 partial products

coef #4 480, 2 partial products

coef #5 128, 1 partial products

coef #6 -48, 2 partial products

coef #7 9, 2 partial products

In total, need 12 partial products

## Matlab code

clear;

addpath('../part1/') %add numppterms

%fir

%coef = [17 -90 241 902 241 -90 17]

%pps 2 4 3 4 3 4 2

% +16 -64 +256 +1024 +256 -64 +16

% +1 -32 -16 -128 -16 -32 +1

% +4 -1 +4 -1 +4

% +2 -2 +2

%original coefficients

coef = [17 -90 241 902 241 -90 17];

%scaling factor

%scale = 1;

scale = 0.532; %12 pps, coef\_r will be [9 -48 128 480 128 -48 9]

%rounded coedficients

coef\_r = round( coef \* scale );

fprintf("Scale %f Using coefficients [", scale);

fprintf(" %d", coef\_r);

fprintf("]\n");

%total number of partial products for FIR

total = 0;

%calculation

for fircoef = 1:7

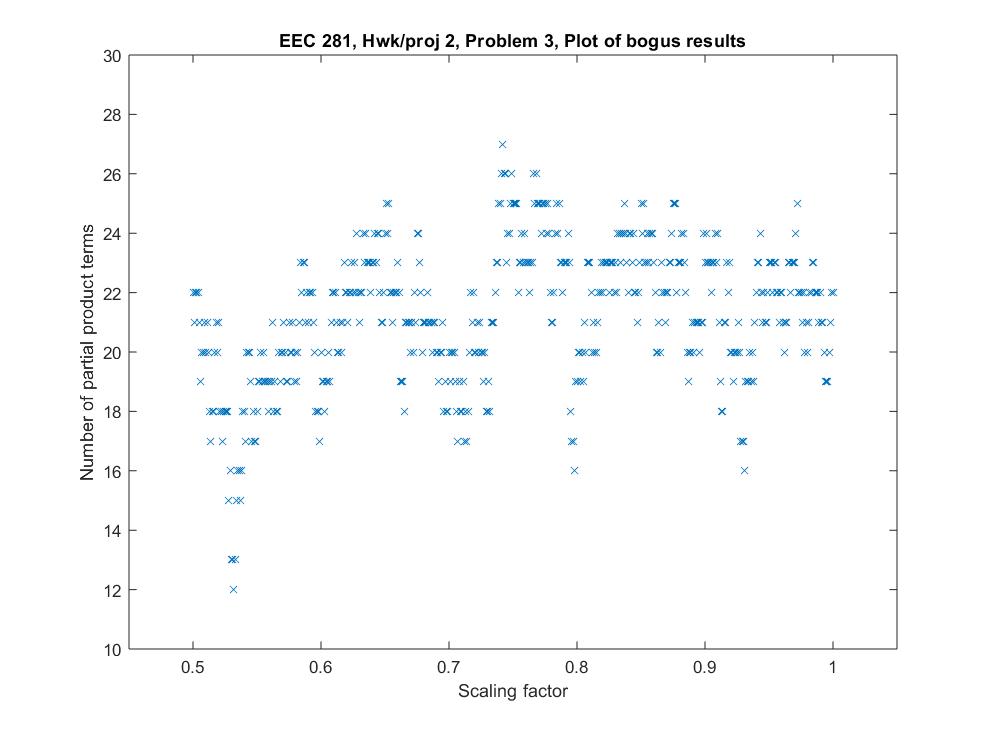
total = total + numppterms(coef\_r(fircoef));

fprintf("coef #%d %d, %d partial products\n", fircoef, coef\_r(fircoef), numppterms(coef\_r(fircoef)));

end

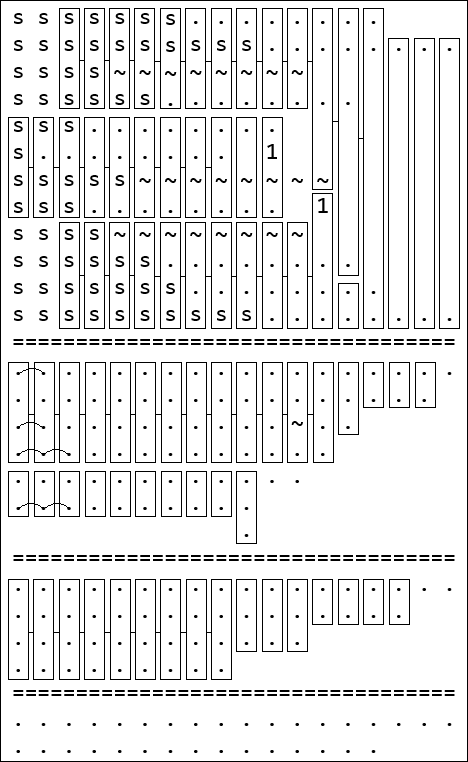
fprintf("In total, need %d partial products\n", total);

## Bogus plot



## Dot diagram

[ ] – 4:2 adder, [ ] – 3:2 adder, [ ] – half adder, ”.” – signal, “~” – inverted signal, “s” – sign extend, ark – right-to-left signal copy to reduce number of adders due to same inputs.



# Problem 3

## Part 1

Energy\_diff / Energy\_data0 = 10\*log(0.000002/4096.020943) = -83.11dB

Difff.m output:

max(data0-data1) = +0.000031 +0.000031i = +3.05034e-05 +3.05034e-05i

min(data0-data1) = -0.000031 -0.000031i = -3.05034e-05 -3.05034e-05i

max(data0/data1) = -Inf -Infi

min(data0/data1) = +Inf +Infi

approx separated mean(data0/data1) = +0.999508+ +0.999508i

Energy\_data0 = 4096.020943

Energy\_diff = 0.000002

Energy\_data0/Energy\_diff = 1653952521.828711 = 92.185230dB

## Part 2

Energy\_diff / Energy\_data0 = 10\*log(0.000002/4096.020943) = -83.113dB

Difff.m output:

max(data0-data1) = +0.000031 +0.000031i = +3.05034e-05 +3.05034e-05i

min(data0-data1) = -0.000031 -0.000031i = -3.05034e-05 -3.05034e-05i

max(data0/data1) = -Inf -Infi

min(data0/data1) = +Inf +Infi

approx separated mean(data0/data1) = +0.999508+ +0.999508i

Energy\_data0 = 4096.020943

Energy\_diff = 0.000002

Energy\_data0/Energy\_diff = 1653952521.828711 = 92.185230dB

## Part 3

Energy\_diff / Energy\_data0 = 10\*log(0.002413/4096.016469) = -62.298dB

Difff.m output:

max(data0-data1) = +0.001554 +0.001554i = +0.00155412 +0.00155412i

min(data0-data1) = -0.001554 -0.001554i = -0.00155412 -0.00155412i

max(data0/data1) = -Inf -Infi

min(data0/data1) = +Inf +Infi

approx separated mean(data0/data1) = +0.997796+ +0.997796i

Energy\_data0 = 4096.016469

Energy\_diff = 0.002413

Energy\_data0/Energy\_diff = 1697518.136491 = 62.298144dB