

---

• Matlab Commands for finding the DFT by DIF-FFT Algorithm

```

x=[2,0,2,0,2,0,2,0];
length=8;%Length of the input
levels=log2(8);%no of multiplication and addition required

input=x; %input to the first level
for i=[3,2,1] %Number of levels in DIF

    %Butterfly Addition and Subtraction
    new_block=[]; %Creating a empty array for output after every level
    for j=1:(2^i):8 %For finding the starting vertices of every block
        block_index=(2^i)-1; % (2^i)-1, For finding the end of each block
        block=input(j:j+block_index); %Selecting a block from above step

        add_index=(2^(i-1))-1; % (2^(level-1))-1 level=i
        add_block=block(1:1+add_index); %Finding the addition block from the
        %block where addition will take place
        diff_block=block(1+add_index+1:2+2*add_index); %Finding the
        %difference block from block where subtraction will take place

        new_add_block=add_block+diff_block; %Additions
        new_diff_block=add_block-diff_block; %Subtractions
        new_block=horzcat(new_block,new_add_block,new_diff_block);
        %Horizontally concatenating the new_block, new_add_block and
        %new_diff_block to create output of each level
    end

    %W_upper for array with value 1 to be multiplied to upper blocks
    W_upper=ones(1, 2^(i-1)); % 2^(level-1) where level=i
    for j=0:( (2^i)/2)-1 %For finding the index for W_lower (j)
        W_found=W(2^i,j);% W(2^i,2^level-number)
        W_lower(j+1)=W_found;%W_lower stored
    end
    Wxxx=horzcat(W_upper,W_lower); %Concatenating the W_upper and W_lower
    %to create a variable Wxxx to be multiplied to a block
    W_upper=[]; % Initializing W_upper to be empty array
    W_lower=[]; % Initializing W_lower to be empty array

    %To multiply W with the blocks
    index_plus=(2^i)-1; % Variable to find till what index the block
    %has to be taken
    for k=1:(2^i):8 %Variable k for finding the starting of the block at
        % level i
        sub_block=new_block(k:k+index_plus); %Finding the block from the input
        x_temp(k:k+index_plus)=sub_block.*Wxxx; % Multiplying W with the block
    end

    input=x_temp; %Output of each level becomes input to next level
end

%Bit reversal

```

---

```

index=0:7; %index value after butterfly addition and subtractions
bit_index=zeros(1,8); %Initializing the variable bit_index
bit_index=bitrevorder(index); % To store the bit reversed index of index

%
for i=1:8
    x_new(i)=x_temp(bit_index(i)+1);
    %Finding the array from bit reverse index of array got from butterfly
    %addition and subtraction
end

xfft_dif=round(x_new);%Rounding of the output fft to find the output fft

```

## Results for Q2

- Plot for the Question No 2

xfft\_dif

*xfft\_dif* =

8      0      0      0      8      0      0      0

*Published with MATLAB® R2014a*