ANNEXURE 5.2

5.2 Innovations by the Faculty in Teaching and Learning

Following are the innovative techniques incorporated by the faculty in the teaching and learning process:

Sl. No.	Name of the Faculty	Subject	Innovation incorporated in Teaching and Learning
1.	Dr. K. Shridhar	Digital Signal Processing	Demonstration of complex DSP concepts through simulation using MATLAB.
2.	Dr. K. Shridhar	Digital Signal Processing	Interactive contents on topic Fast Fourier Transform. https://docs.google.com/presentation/d/1WfYMCg7GBVueFVwJVROS1vEqEAueqKjc/edit?usp=drive_link&ouid=101099264730601342587&rtpof=true&sd=true
3.	Dr. Jayashree D. Mallapur	Multimedia Communication	Assignments for the subject multimedia communication handled by the faculty. https://classroom.google.com/c/NjQyODQ4Nzg1NzUy/p/NjUwNTIyNjQxNTk3/details
4.	Dr. Nagarathna Rajur	Internet of Things	Testing models knowledge. https://forms.gle/qGRRW723Q7S3JAW76
5.	Dr. Vijaylakshmi Jigajinni	Image Processing	Self evaluation by students after their presentation: as part of assignment A rubrics is prepared for evaluating the student performance after their presentations and they are made to fill the goggle forms by themselves so that they can find out by themselves how they presented the assignment.
6.	Dr. Ashok Sutagundar	Computer Networks	Power point presentations, Demonstrations of network equipments in the classes
7.	Dr. S. G. Kambalimath	Verilog Programming	Website is used for teaching Verilog Programming as additional material. http://www.chipverify.com/
8.	Dr. S. G. Kambalimath	UEC543C: Verilog Programming	Project-Based Learning (PBL)
9.	Dr. S. G. Kambalimath	Digital System Design using Verilog	Simulations
10.	Dr. S. G. Kambalimath	UEC543C: Verilog Programming	Self-Paced Learning
11.	Dr. Vijaylakshmi S. Jigajinni	Aircraft electronics system	TEACHER FEEDBACK ON CURRICULUM AND INFRASTRUCTURE 2023-24https://docs.google.com/forms/d/1f1o7tGDmVF8Y9-VN2eKsePIqa6gvPqRLrkFVM2XQZ5A/viewform?edit_requested=true
12.	Dr. Vijaylakshmi S. Jigajinni	Aircraft electronics and system	Presentation by group of students on topic given. The evaluation of the presentation was done using

			the Google spread sheets. Link for the same is as given below. https://docs.google.com/spreadsheets/d/1pjcFe7E9s3TGEiK_fUCeeYlm31PJFr9czwkxwtmcxM/edit?usp=drivesdk
13.	Dr. Viiavlakshmi S. Jigajinni	Aircraft electronics and system	Smart board utilization
14.	Mamata J Sataraddi	Network Analysis	Practical are done to clarify the theory concepts
15.	Dr. Ashok Sutagundar	Data Structure using C	Power point presentations, Simulation of the programs through online classes
16.	Dr. Ashok Sutagundar	Mobile Communications	Power point presentations, Demonstrations of network equipments in the classes

HOD

Professor and Head
Department of Electronics & Communication Engg
Basaveshwar Engineering College,
BAGALKOT-587102

BVVS
Basaveshwar Engineering College, Bagalkote
Department of Electronics and Communication Engineering

Fast Fourier Transform (FFT) Algorithms

Dr. K. Shridhar

Relation to the z-transform

$$x(n) = \begin{cases} \text{Non zero}, & 0 \le n \le N-1 \\ 0, & \text{elsewhere} \end{cases} \qquad X(z) = \sum_{n=0}^{N-1} x(n) z^{-n}$$

$$x(n) = \begin{cases} \tilde{x}(n), & 0 \le n \le N-1 \\ 0, & \text{elsewhere} \end{cases} \qquad \tilde{X}(k) = \sum_{n=0}^{N-1} x(n) \left[e^{j\frac{2\pi}{N}k}\right]^{-n}$$

$$\tilde{X}(k) = X(z)\Big|_{z=e^{j\frac{2\pi}{N}k}}$$

The DFT, X(k) represents N equally spaced samples of z-transform X(z), on the circumference of unit circle.

Relation to the DTFT

$$X(e^{jw}) = \sum_{n=0}^{N-1} x(n) e^{-jwn} = \sum_{n=0}^{N-1} \tilde{x}(n) e^{-jwn}$$

$$\tilde{X}(k) = X(e^{jw})\big|_{w=\frac{2\pi}{N}k}$$

$$Let \quad w_1 = \frac{2\pi}{N}, \ and \ w_k = \frac{2\pi}{N}k = kw_1$$

$$X(k) = X(e^{jw_k}) = X(e^{jkw_1})$$

DFT is obtained by uniformly sampling the DTFT at w_1 intervals

The interval w_1 is the *sampling interval* in the frequency domain. It is called *frequency resolution* because it tells us the minimum frequency quantum of information which we can have about the signal. Smaller the value better the resolution.

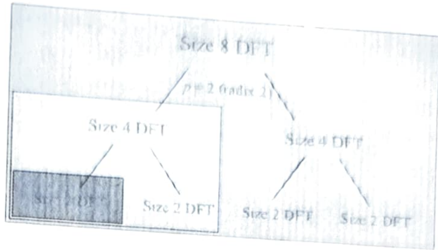
Comments

- When we sample X(z) on the circumference of unit circle, we obtain a periodic sequence in the time domain.
- This sequence is a linear combination of the original x(n) and its infinite replicas, each shifted by multiples of N or –N samples.
- If x(n)=0 for n<0 and n>=N, then there will be no overlap or aliasing in the time domain.

Important Points

- Zero-padding is an operation in which extra zeros are appended to the original sequence at the end of the signal. The resulting longer DFT provides closely spaced samples of the discrete Fourier transform of the original sequence.
- The zero-padding gives us a high-density spectrum and provides a better display of the spectrum. But it does not provide high-resolution spectrum because no new information is added to the signal; only additional zeros are inserted in the data.
- To get high-resolution spectrum, we need more data.
- More the data, more the information

Divide & Conquer Method



Important Computational Requirements

- Total number of computations should be linear function rather than quadratic function of N.
- Most of the computations can be eliminated using the symmetry and periodicity properties of Twiddle Factors

$$W_N^{kn} = W_N^{k(n+N)} = W_N^{(k+N)n}$$
$$W_N^{kn+N/2} = -W_N^{kn}$$

Decimation-in-time FFT Algorithm: DIT-FFT Algorithm

Decimation-in-frequency FFT Algorithm: DIF-FFT Algorithm

Radix-2 FFT Algorithms (DIT-FFT Algorithm)

- Let $N=2^v$; then we choose M=2 and L=N/2 and divide x(n) into two N/2-point sequence.
- Above step is repeated. At each stage the sequences are decimated and the smaller DFTs are combined. This decimation ends after v stages.
- The resulting procedure is called Decimation-in-Time FFT (DIF-FFT) algorithm, for which the total number of complex multiplications is: C_N=Nv= N*$\log_2$N;
- Using additional symmetries: C_N=Nv= N/2*$\log_2$N

Radix-2 FFT Algorithms (DIF-FFT Algorithm)

- Choose L=2, M=N/2 and follow the steps of DIT-FFT algorithm.
- Its signal flow graph is transposed structure of the DIT-FFT structure.
- Its computational complexity is also equal to $C_N = Nv = N/2 * \log_2 N$

Merits of Radix-2 FFT Algorithms

- To reduce computational complexity and increase computational efficiency, it is necessary to decompose N point DFT computation into successively smaller (N/2, N/4, N/8, N/16, ...) DFT computations. In this process we have to exploit both symmetry and periodicity property of complex exponentials

$$W_N^{kn} = W_N^{k(n+N)} = W_N^{(k+N)n}$$
$$W_N^{kn+N/2} = -W_N^{kn}$$

Decimation-in-time FFT Algorithm

- The principle of decimation-in-time is most conveniently illustrated by considering the special case of Let N = An integer power of 2: $N=2^v$
- Since N is an even integer, we can consider computing X(k) by separating x(n) into two N/2-point sequences consisting of the even-indexed points and odd-indexed points in x(n). Mathematically X(k) is given by

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \qquad (2.3)$$

- Separating x(n) into even-numbered and odd-numbered points, we get new expression for X(k)

$$X(k) = \sum_{n \, even} x(n) W_N^{nk} + \sum_{n \, odd} x(n) W_N^{nk}$$

Now substitute n=2r for an even and n=2r+1 for odd indexed samples

$$X(k) = \sum_{r=0}^{\frac{N}{2}-1} x(2r) W_N^{2rk} + \sum_{r=0}^{\frac{N}{2}-1} x(2r+1) W_N^{(2r+1)k}$$
$$= \sum_{r=0}^{\frac{N}{2}-1} x(2r) (W_N^2)^{rk} + W_N^k \sum_{r=0}^{\frac{N}{2}-1} x(2r+1) (W_N^2)^{rk} \qquad (2.4)$$

But we know that $W_N^2 = W_{N/2}$

As a result previous equation (2.4) can be written as

$$X(k) = \sum_{r=0}^{\frac{N}{2}-1} x(2r) W_{\frac{N}{2}}^{rk} + W_N^k \sum_{r=0}^{\frac{N}{2}-1} x(2r+1) W_{\frac{N}{2}}^{rk}$$

$$= G(k) + W_N^k H(k) \qquad k = 0, 1, \ldots, N-1 \qquad (2.5)$$

Each of the sums in equation (2.5) is recognized as an N/2-point DFT. Each of sums need only be computed for k = 0 to N/2 to give G(k) and H(k). Since each G(k) and H(k) are periodic in k with period N/2 further they can be decomposed into two N/4 point DFTs. This process should be continued till we reach at two point DFTs.

The computational flow or the signal flow in computing X(k) according to Eq. (2.5) for an 8-point sequence, i.e. N=8 is shown in Figure below.

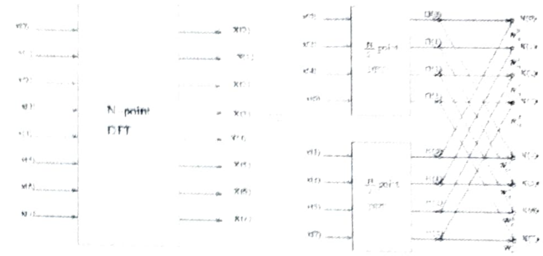


Figure 2.1: Flow graph of the decimation-in-time decomposition of an N ($N = 8$) point DFT into $\frac{N}{2}$-point DFT

- Further G(k) and H(k) in equation (2.5) can be computed as indicated in the next slide

$$G(k) = \sum_{r=0}^{\frac{N}{2}-1} g(r) W_{\frac{N}{2}}^{rk}$$

$$= \sum_{l=0}^{\frac{N}{4}-1} g(2l) W_{\frac{N}{2}}^{2lk} + \sum_{l=0}^{\frac{N}{4}-1} g(2l+1) W_{\frac{N}{2}}^{(2l+1)k}$$

$$= \sum_{l=0}^{\frac{N}{4}-1} g(2l) W_{\frac{N}{4}}^{lk} + W_{\frac{N}{2}}^{k} \sum_{l=0}^{\frac{N}{4}-1} g(2l+1) W_{\frac{N}{4}}^{lk} \qquad (2.6)$$

On the same lines

$$H(k) = \sum_{l=0}^{\frac{N}{4}-1} h(2l) W_{\frac{N}{4}}^{lk} + W_{\frac{N}{2}}^{k} \sum_{l=0}^{\frac{N}{4}-1} h(2l+1) W_{\frac{N}{4}}^{lk} \qquad (2.7)$$

Where g(r)=x(2r) and h(r)=h(2r+1)

If the 4-point DFTs in Figure (2.1) are computed according to equations (2.6) and (2.7), then those computations would be carried out as indicated in Figure (2.2) drawn below

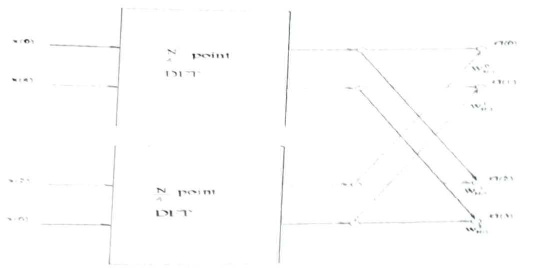


Figure 2.2: Flow graph of decimation-in-time decomposition of an $\frac{N}{2}$-point DFT into $\frac{N}{4}$-point DFT ($N = 8$).

Inserting the computation indicated in Figure (2.2) into the flow graph of Figure (2.1), we obtain the complete flow graph of Figure (2.3). We have used the fact that $W_{N/2} = (W_N)^2$.

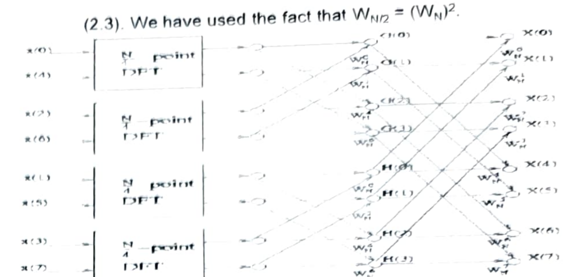


Figure 2.3: Result of substitution of flow graph of Figure (2.1) into Flow graph of Figure (2.1).

In-place Computations

- In view of Figure (2.4), the Figure (2.3) gives the complete computational flow graph for the N-point computation of DFT of N-point sequence, for N=8.
- An interesting by-product of this derivation is that, this flow graph, in addition to describing efficient procedure for computing the DFT, also suggests a useful way of storing the original data and storing the results of the computation in the intermediate arrays.

For N=8, N/4-point DFT becomes 2-point DFT. The 2-point DFT of, for example, x(0) and x(4) is depicted in Figure (2.4).

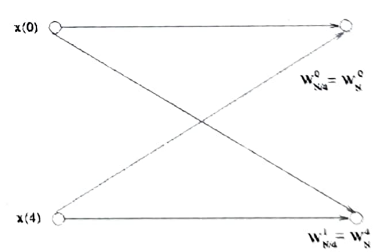


Figure 2.4: Flow graph of a two-point DFT (for $N = 8$)

We shall denote the sequence of complex numbers resulting from the m^{th} stage of computation as $X_m(l)$, where l=0,1,...,N-1, and m=1,2,..., forming an input to the $(m+1)^{st}$ stage and producing an output $X_{m+1}(l)$ as the output from the $(m+1)^{st}$ stage of computations, it can be seen that the basic computation in flow graph of Figure (2.3)

Figure 2.5: Flow graph of basic butterfly computation in Figure (2.3)

The equations represented by this flow graph are

$$\begin{aligned} X_{m+1}(p) &= X_m(p) + W_N^r X_m(q) \\ X_{m+1}(q) &= X_m(p) + W_N^{r+\frac{N}{2}} X_m(q) \end{aligned} \qquad (2.8)$$

Because of the appearance of the flow graph of Figure (2.5), this computation is referred as a butterfly computation.

Equations (2.8) suggest a means of reducing the number of complex multiplications by a factor of 2. To see this we note that

$$W_N^{\frac{N}{2}} = e^{-j\frac{2\pi}{N}\frac{N}{2}} = e^{-j\pi} = -1$$

So the equations (2.8) become

$$\begin{aligned} X_{m+1}(p) &= X_m(p) + W_N^r X_m(q) \\ X_{m+1}(q) &= X_m(p) - W_N^r X_m(q) \end{aligned} \qquad (2.9)$$

Equations (2.9) are represented in the flow graph of Figure (2.6).

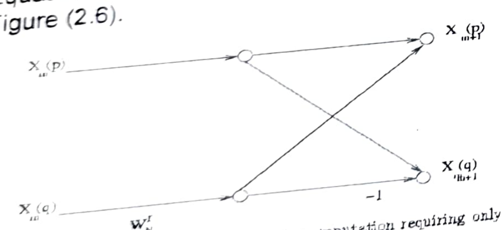


Figure 2.6: Flow graph of simplified butterfly computation requiring only one complex multiplication

Combining the observations in Figures (2.6), (2.5), (2.4) and (2.3), the efficient FFT algorithm in the computational flow graph representation for N=8 is obtained as shown in Figure (2.7). The algorithm requires $N/2 \log_2 N$ complex multiplications and $N \log_2$

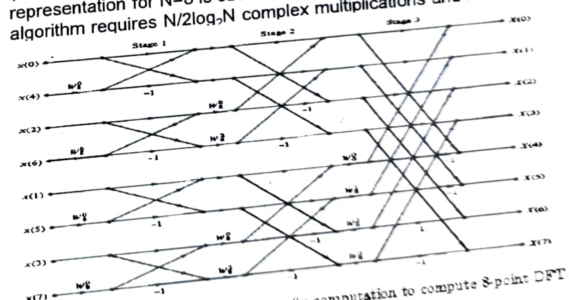


Figure 2.7: Complete flow graph of butterfly computation to compute 8-point DFT

Decimation-in-Frequency FFT Algorithm

- The decimation-in-time FFT algorithms were all based upon the decomposition of the DFT computation by forming smaller and smaller subsequences.
- Alternatively decimation-in-frequency FFT algorithms are all based upon decomposition of the DFT computation over X(k). For N a power of 2 i.e.

$$N = 2^{\nu}$$

we divide the input sequence into first half and the last half of points so that

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} x(n) W_N^{nk} + \sum_{n=\frac{N}{2}}^{N-1} x(n) W_N^{nk}, \qquad k = 0, 1, \ldots, N-1$$

or

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} x(n) W_N^{nk} + W_N^{\frac{N}{2}k} \sum_{n=0}^{\frac{N}{2}-1} x(n + \frac{N}{2}) W_N^{nk}$$

$$= \sum_{n=0}^{\frac{N}{2}-1} [x(n) + (-1)^k x(n + \frac{N}{2})] W_N^{nk} \qquad \text{since, } W_N^{\frac{N}{2}k} = (-1)^k \qquad (2.10)$$

- Separating k-even and k-odd, i.e. k=2r and k=2r+1, representing the even-numbered points and the odd-numbered points, respectively, so that

$$X(2r) = \sum_{n=0}^{\frac{N}{2}-1} [x(n) + x(n + \frac{N}{2})] W_{\frac{N}{2}}^{nr}, \qquad r = 0, 1, \ldots, \frac{N}{2} - 1 \quad (2.11)$$

$$X(2r+1) = \sum_{n=0}^{\frac{N}{2}-1} [x(n) - x(n + \frac{N}{2})] W_N^n W_{\frac{N}{2}}^{nr}, \qquad r = 0, 1, \ldots, \frac{N}{2} - 1 \quad (2.12)$$

Thus on the basis of Equations (2.11) and (2.12) with

$$g(n) = x(n) + x(n + \frac{N}{2})$$

and

$$h(n) = x(n) - x(n + \frac{N}{2})$$

The DFT can be computed by first forming the sequences g(n) and h(n), then computing $h(n)W_N^n$, and finally computing the N/2-point DFTs of these two sequences to obtain the even-numbered output points and odd-numbered output points, respectively.

The procedure suggested by Eqs. (2.11) and (2.12) is illustrated through signal flow graph for the case of 8-point DFT in Figure (2.8).

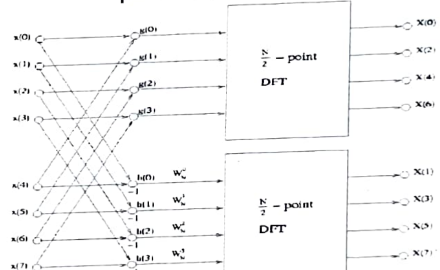


Figure 2.8: Flow graph of the decimation-in-frequency decomposition of an N-point computation into two $\frac{N}{2}$-point DFT computations, for $N = 8$

Proceeding in a manner similar to that followed in deriving the decimation-in-time algorithm, the final signal flow graph for computation is shown in Figure (2.9).

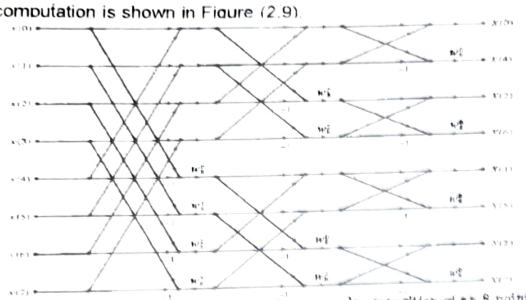


Figure 2.9: Flow graph of the decimation-in-frequency decomposition of an 8-point DFT computation

- By counting the arithmetic operations in Figure (2.9), and generalizing, we see that the computation of Figure (2.9) requires $N/2\log_2 N$ complex multiplications and $N\log_2 N$ complex additions. Thus the total computation is the same for decimation-in-frequency and decimation-in-time algorithms.
- Similar to decimation-in-time algorithm the computational flow graph shown in Figure (2.9) will indicate the in-place computation capability of decimation-in-frequency algorithm.
- Figure (2.9) is the transpose of Figure (2.7).

Decimation-In-Time FFT Algorithms

- Makes use of both symmetry and periodicity
- Consider special case of N an integer power of 2
- Separate x[n] into two sequence of length N/2
  - Even indexed samples in the first sequence
  - Odd indexed samples in the other sequence

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi/Nkn} = \sum_{n \text{ even}}^{N-1} x[n]e^{-j2\pi/Nkn} + \sum_{n \text{ odd}}^{N-1} x[n]e^{-j2\pi/Nkn}$$

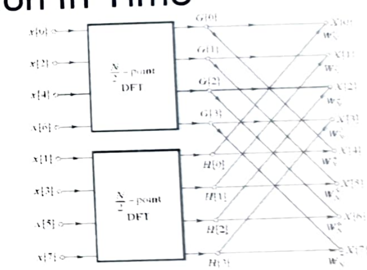
- Substitute variables n=2r for n even and n=2r+1 for odd

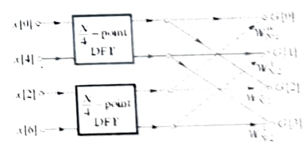
$$\begin{aligned} X[k] &= \sum_{r=0}^{N/2-1} x[2r]W_N^{2rk} + \sum_{r=0}^{N/2-1} x[2r+1]W_N^{(2r+1)k} \\ &= \sum_{r=0}^{N/2-1} x[2r]W_{N/2}^{rk} + W_N^k \sum_{r=0}^{N/2-1} x[2r+1]W_{N/2}^{rk} \\ &= G[k] + W_N^k H[k] \end{aligned}$$

- G[k] and H[k] are the N/2-point DFT's of each subsequence

Decimation In Time

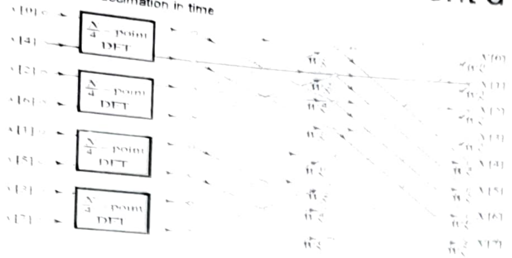
- 8-point DFT example using decimation-in-time
- Two N/2-point DFTs
  - $2(N/2)^2$ complex multiplications
  - $2(N/2)^2$ complex additions
- Combining the DFT outputs
  - N complex multiplications
  - N complex additions
- Total complexity
  - $N^2/2+N$ complex multiplications
  - $N^2/2+N$ complex additions
  - More efficient than direct DFT
- Repeat same process
  - Divide N/2-point DFTs into
  - Two N/4-point DFTs
  - Combine outputs





Decimation In Time Cont'd

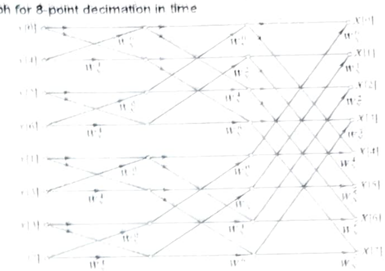
- After two steps of decimation in time



- Repeat until we're left with two-point DFT's

Decimation-In-Time FFT Algorithm

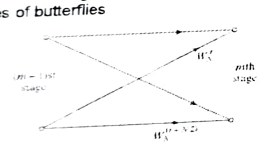
- Final flow graph for 8-point decimation in time



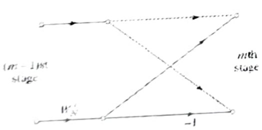
- Complexity
  - $N\log_2 N$ complex multiplications and additions

Butterfly Computation

- Flow graph constitutes of butterflies



- We can implement each butterfly with one multiplication



- Final complexity for decimation-in-time FFT
  - $(N/2)\log_2 N$ complex multiplications and additions

In-Place Computation

- Decimation-in-time flow graphs require two sets of registers
  - Input and output for each stage
- Note the arrangement of the input indices
  - Bit reversed indexing

$$X_0[0] = x[0] \leftrightarrow X_0[000] = x[000]$$
$$X_0[1] = x[4] \leftrightarrow X_0[001] = x[100]$$
$$X_0[2] = x[2] \leftrightarrow X_0[010] = x[010]$$
$$X_0[3] = x[6] \leftrightarrow X_0[011] = x[110]$$
$$X_0[4] = x[1] \leftrightarrow X_0[100] = x[001]$$
$$X_0[5] = x[5] \leftrightarrow X_0[101] = x[101]$$
$$X_0[6] = x[3] \leftrightarrow X_0[110] = x[011]$$
$$X_0[7] = x[7] \leftrightarrow X_0[111] = x[111]$$

Decimation-In-Frequency FFT Algorithm

- The DFT equation

$$X[k] = \sum_{n=0}^{N-1} x[n]W_N^{nk}$$

- Split the DFT equation into even and odd frequency indexes

$$X[2r] = \sum_{n=0}^{N-1} x[n]W_N^{n2r} = \sum_{n=0}^{N/2-1} x[n]W_N^{n2r} + \sum_{n=N/2}^{N-1} x[n]W_N^{n2r}$$

- Substitute variables to get

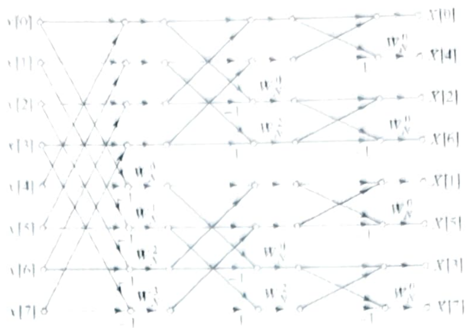
$$X[2r] = \sum_{n=0}^{N/2-1} x[n]W_N^{n2r} + \sum_{n=0}^{N/2-1} x[n+N/2]W_N^{(n+N/2)2r} = \sum_{n=0}^{N/2-1} (x[n] + x[n+N/2])W_{N/2}^{nr}$$

- Similarly for odd-numbered frequencies

$$X[2r+1] = \sum_{n=0}^{N/2-1} (x[n] - x[n+N/2])W_N^{n(2r+1)}$$

Decimation-In-Frequency FFT Algorithm

- Final flow graph for 8-point decimation in frequency



FFT vs. DFT

- The FFT is simply an algorithm for efficiently calculating the DFT
- Computational efficiency of an N-Point FFT:
  - DFT: N^2 Complex Multiplications
  - FFT: $(N/2)\log_2(N)$ Complex Multiplications

N	DFT Multiplications	FFT Multiplications	FFT Efficiency
256	65,536	1,024	64 : 1
512	262,144	2,304	114 : 1
1,024	1,048,576	5,120	205 : 1
2,048	4,194,304	11,264	372 : 1
4,096	16,777,216	24,576	683 : 1

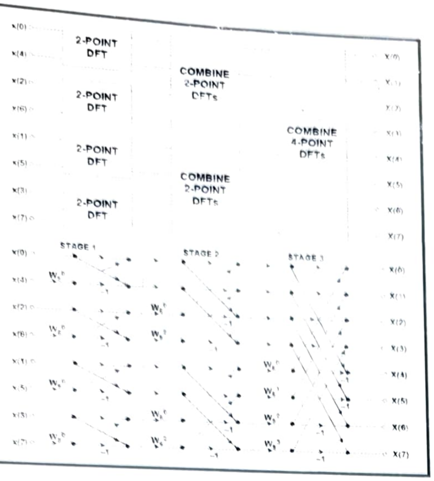
Bit Reversal

Decimal Number :	0	1	2	3	4	5	6	7
Binary Equivalent :	000	001	010	011	100	101	110	111
Bit-Reversed Binary :	000	100	010	110	001	101	011	111
Decimal Equivalent :	0	4	2	6	1	5	3	7

- The *bit reversal* algorithm used to perform the re-ordering of signals.
- The decimal index, n, is converted to its binary equivalent.
- The binary bits are then placed in reverse order, and converted back to a decimal number.
- Bit reversing is often performed in DSP hardware in the data address generator (DAG).

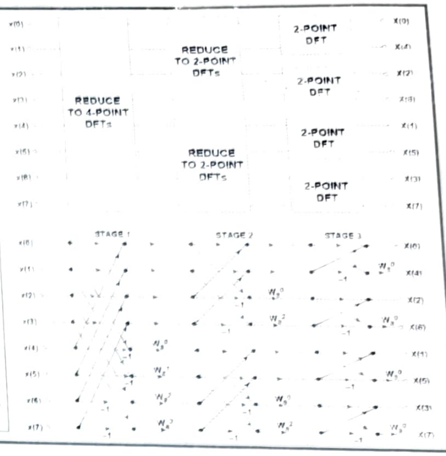
- Input signal must be properly re-ordered using a *bit reversal* algorithm
- In-place computation
- Number of stages: $log_2 N$
- Stage 1: all the twiddle factors are 1
- Last Stage: the twiddle factors are in sequential order

	Stage 1	Stage 2	Stage 3	Stage Log_2N
Number of Groups	N/2	N/4	N/8	1
Butterflies per Group	1	2	4	N/2
Dual-Node Spacing	1	2	4	N/2
Twiddle Factor Exponents	(N/2)k, k=0	(N/4)k, k=0,1	(N/8)k, k=0,1,2,3	k, k=0 to N/2−1

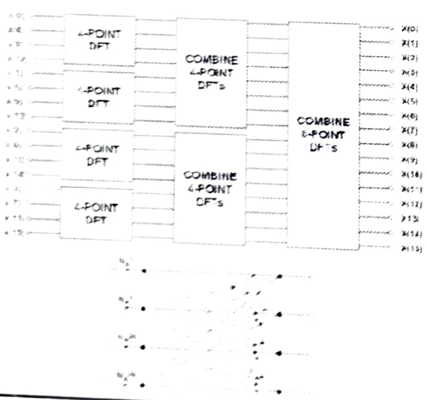


- Output signal must be properly re-ordered using a *bit reversal* algorithm
- In-place computation
- Number of stages: $log_2 N$
- Stage 1: the twiddle factors are in sequential order
- Last Stage: all the twiddle factors are 1

	Stage 1	Stage 2	Stage 3	Stage Log_2N
Number of Groups	1	2	4	N/2
Butterflies per Group	N/2	N/4	N/8	1
Dual-Node Spacing	N/2	N/4	N/8	1
Twiddle Factor Exponents	n, n=0 to N/2 - 1	2n, n=0 to N/4 - 1	4n, n=0 to N/8 - 1	(N/2)n, n=0



Radix-4 Decimation-In-Time FFT Algorithm



- A radix-4 FFT combines two stages of a radix-2 FFT into one, so that half as many stages are required.
- The radix-4 butterfly is consequently larger and more complicated than a radix-2 butterfly.
- N/4 butterflies are used in each of $(log_2N)/2$ stages, which is one quarter the number of butterflies in a radix-2 FFT.
- Addressing of data and twiddle factors is more complex, a radix-4 FFT requires fewer calculations than a radix-2 FFT.
- It can compute a radix-4 FFT significantly faster than a radix-2 FFT.

Inverse Discrete Fourier Transform (IDFT)

The inverse discrete Fourier transform (IDFT) is given by

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k) W_N^{-kn}, \qquad n = 0, 1, \ldots, N-1 \qquad (2.13)$$

which is structurally similar to DFT,

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \qquad (2.14)$$

The change we notice is in the multiplication factor 1/N} and replacement of W_N^{kn} by W_N^{-kn}, and the interchange of signals x(n) and X(k) in the expressions and the index for summation.

- Thus in Figure (2.7) and (2.9), if we exercise the above changes, the changed signal flow graphs will become algorithms for IDFT and referred as IFFT algorithms.

Example

- Using decimation-in-time FFT algorithm compute DFT of the sequence

  {-1 –1 –1 –1 1 1 1 1}

- Solution: Twiddle factors are

$$W_8 = e^{-j\frac{2\pi}{8}} = e^{-j\frac{\pi}{4}} = 0.707 - j0.707$$

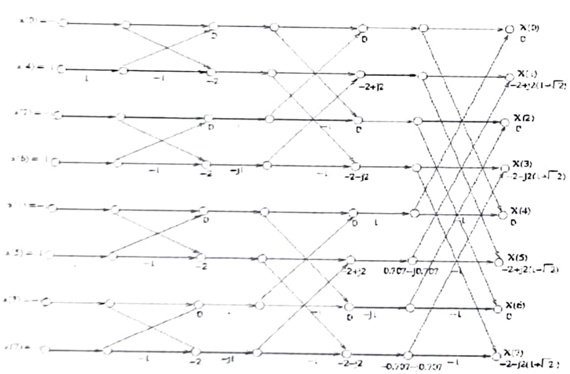
$$W_8^0 = 1$$

$$W_8^1 = 0.707 - j0.707$$

$$W_8^2 = -j1$$

$$W_8^3 = -0.707 - j0.707$$

Solution and signal flow graph of the example



Thank You

BVVS

Basaveshwar Engineering College, Bagalkote

Department of Electronics and Communication Engineering

Fast Fourier Transform (FFT) Algorithms

Dr. K. Shridhar

Relation to the z-transform

$$x(n) = \begin{cases} \text{Non zero}, & 0 \le n \le N-1 \\ 0, & \text{elsewhere} \end{cases} \qquad X(z) = \sum_{n=0}^{N-1} x(n) z^{-n}$$

$$x(n) = \begin{cases} \tilde{x}(n), & 0 \le n \le N-1 \\ 0, & \text{elsewhere} \end{cases} \qquad \tilde{X}(k) = \sum_{n=0}^{N-1} x(n) \left[e^{j\frac{2\pi}{N}k}\right]^{-n}$$

$$\tilde{X}(k) = X(z)\big|_{z=e^{j\frac{2\pi}{N}k}}$$

The DFT, X(k) represents N equally spaced samples of z-transform X(z), on the circumference of unit circle.

Relation to the DTFT

$$X(e^{jw}) = \sum_{n=0}^{N-1} x(n) e^{-jwn} = \sum_{n=0}^{N-1} \tilde{x}(n) e^{-jwn}$$

$$\tilde{X}(k) = X(e^{jw})\big|_{w=\frac{2\pi}{N}k}$$

$$Let \quad w_1 = \frac{2\pi}{N}, \; and \; w_k = \frac{2\pi}{N}k = kw_1$$

$$X(k) = X(e^{jw_k}) = X(e^{jkw_1})$$

DFT is obtained by uniformly sampling the DTFT at w_1 intervals

The interval w_1 is the *sampling interval* in the frequency domain. It is called *frequency resolution* because it tells us the minimum frequency quantum of information which we can have about the signal. Smaller the value better the resolution.

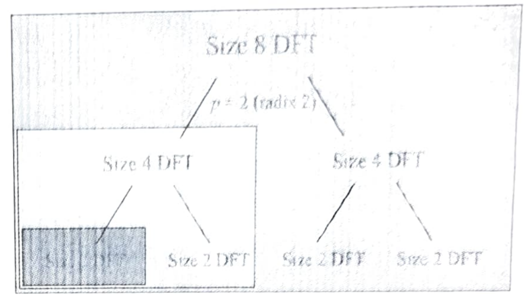
Comments

- When we sample X(z) on the circumference of unit circle, we obtain a periodic sequence in the time domain.
- This sequence is a linear combination of the original x(n) and its infinite replicas, each shifted by multiples of N or –N samples.
- If x(n)=0 for n<0 and n>=N, then there will be no overlap or aliasing in the time domain.

Important Points

- Zero-padding is an operation in which extra zeros are appended to the original sequence at the end of the signal. The resulting longer DFT provides closely spaced samples of the discrete Fourier transform of the original sequence.
- The zero-padding gives us a high-density spectrum and provides a better display of the spectrum. But it does not provide high-resolution spectrum because no new information is added to the signal; only additional zeros are inserted in the data.
- To get high-resolution spectrum, we need more data.
- More the data, more the information

Divide & Conquer Method



Important Computational Requirements

- Total number of computations should be linear function rather than quadratic function of N.
- Most of the computations can be eliminated using the symmetry and periodicity properties of Twiddle Factors

$$W_N^{kn} = W_N^{k(n+N)} = W_N^{(k+N)n}$$

$$W_N^{kn+N/2} = -W_N^{kn}$$

Decimation-in-time FFT Algorithm: DIT-FFT Algorithm

Decimation-in-frequency FFT Algorithm: DIF-FFT Algorithm

Radix-2 FFT Algorithms (DIT-FFT Algorithm)

- Let $N=2^v$; then we choose M=2 and L=N/2 and divide x(n) into two N/2-point sequence.
- Above step is repeated. At each stage the sequences are decimated and the smaller DFTs are combined. This decimation ends after v stages.
- The resulting procedure is called Decimation-in-Time FFT (DIF-FFT) algorithm, for which the total number of complex multiplications is: $C_N=Nv= N*\log_2 N$;
- Using additional symmetries: $C_N=Nv= N/2*\log_2 N$

Radix-2 FFT Algorithms (DIF-FFT Algorithm)

- Choose L=2, M=N/2 and follow the steps of DIT-FFT algorithm.
- Its signal flow graph is transposed structure of the DIT-FFT structure.
- Its computational complexity is also equal to $C_N = N\nu = N/2 * \log_2 N$

Merits of Radix-2 FFT Algorithms

- To reduce computational complexity and increase computational efficiency, it is necessary to decompose N point DFT computation into successively smaller (N/2, N/4, N/8, N/16, ...) DFT computations. In this process we have to exploit both symmetry and periodicity property of complex exponentials

$$W_N^{kn} = W_N^{k(n+N)} = W_N^{(k+N)n}$$
$$W_N^{kn+N/2} = -W_N^{kn}$$

Decimation-in-time FFT Algorithm

- *The principle of decimation-in-time is most conveniently illustrated by considering the special case of Let N = An integer power of 2: N=2^ν*
- *Since N is an even integer, we can consider computing X(k) by separating x(n) into two N/2-point sequences consisting of the even-indexed points and odd-indexed points in x(n). Mathematically X(k) is given by*

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \qquad (2.3)$$

- Separating x(n) into even-numbered and odd-numbered points, we get new expression for X(k)

$$X(k) = \sum_{n \text{ even}} x(n) W_N^{nk} + \sum_{n \text{ odd}} x(n) W_N^{nk}$$

Now substitute n=2r for an even and n=2r+1 for odd indexed samples

$$X(k) = \sum_{r=0}^{\frac{N}{2}-1} x(2r) W_N^{2rk} + \sum_{r=0}^{\frac{N}{2}-1} x(2r+1) W_N^{(2r+1)k}$$
$$= \sum_{r=0}^{\frac{N}{2}-1} x(2r) (W_N^2)^{rk} + W_N^k \sum_{r=0}^{\frac{N}{2}-1} x(2r+1) (W_N^2)^{rk} \qquad (2.4)$$

But we know that $W_N^2 = W_{N/2}$

As a result previous equation (2.4) can be written as

$$X(k) = \sum_{r=0}^{\frac{N}{2}-1} x(2r) W_{\frac{N}{2}}^{rk} + W_N^k \sum_{r=0}^{\frac{N}{2}-1} x(2r+1) W_{\frac{N}{2}}^{rk}$$
$$= G(k) + W_N^k H(k), \qquad k = 0, 1, \ldots, N-1 \qquad (2.5)$$

Each of the sums in equation (2.5) is recognized as an N/2-point DFT. Each of sums need only be computed for k = 0 to N/2 to give G(k) and H(k). Since each G(k) and H(k) are periodic in k with period N/2 further they can be decomposed into two N/4 point DFTs. This process should be continued till we reach at two point DFTs.

The computational flow or the signal flow in computing X(k) according to Eq. (2.5) for an 8-point sequence, i.e. N=8 is shown in Figure below.

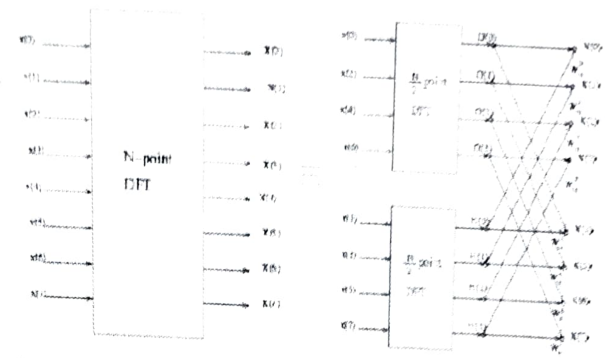


Figure 2.1: Flow graph of the decimation-in-time decomposition of an N ($N = 8$) point DFT into $\frac{N}{2}$-point DFT.

- Further G(k) and H(k) in equation (2.5) can be computed as indicated in the next slide

$$G(k) = \sum_{r=0}^{\frac{N}{2}-1} g(r) W_{\frac{N}{2}}^{rk}$$
$$= \sum_{l=0}^{\frac{N}{4}-1} g(2l) W_{\frac{N}{2}}^{2lk} + \sum_{l=0}^{\frac{N}{4}-1} g(2l+1) W_{\frac{N}{2}}^{(2l+1)k}$$
$$= \sum_{l=0}^{\frac{N}{4}-1} g(2l) W_{\frac{N}{4}}^{lk} + W_{\frac{N}{2}}^{k} \sum_{l=0}^{\frac{N}{4}-1} g(2l+1) W_{\frac{N}{4}}^{lk} \qquad (2.6)$$

On the same lines

$$H(k) = \sum_{l=0}^{\frac{N}{4}-1} h(2l) W_{\frac{N}{4}}^{lk} + W_{\frac{N}{2}}^{k} \sum_{l=0}^{\frac{N}{4}-1} h(2l+1) W_{\frac{N}{4}}^{lk} \qquad (2.7)$$

Where g(r)=x(2r) and h(r)=h(2r+1)

If the 4-point DFTs in Figure (2.1) are computed according to equations (2.6) and (2.7), then those computations would be carried out as indicated in Figure (2.2) drawn below

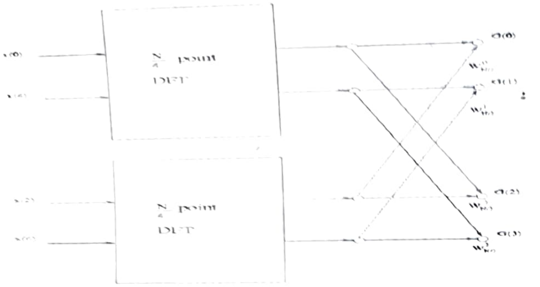


Figure 2.2: Flow graph of decimation-in-time decomposition of an $\frac{N}{2}$-point DFT into $\frac{N}{4}$-point DFT ($N = 8$).

Inserting the computation indicated in Figure (2.2) into the flow graph of Figure (2.1), we obtain the complete flow graph of Figure (2.3). We have used the fact that $W_{N/2} = (W_N)^2$.

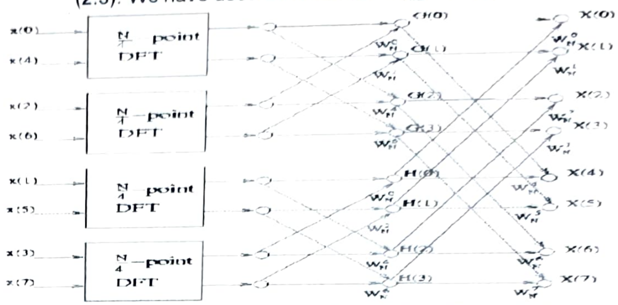


Figure 2.3: Result of substitution of flow graph of Figure (2.1) into Flow graph of Figure (2.1).

In-place Computations

- In view of Figure (2.4), the Figure (2.3) gives the complete computational flow graph for the N-point computation of DFT of N-point sequence, for N=8.
- An interesting by-product of this derivation is that, this flow graph, in addition to describing efficient procedure for computing the DFT, also suggests a useful way of storing the original data and storing the results of the computation in the intermediate arrays.

For N=8, N/4-point DFT becomes 2-point DFT. The 2-point DFT of, for example, x(0) and x(4) is depicted in Figure (2.4).

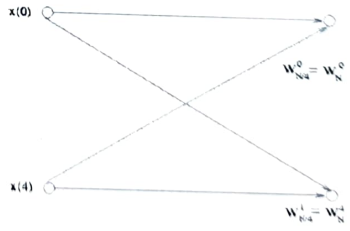


Figure 2.4: Flow graph of a two-point DFT (for $N = 8$)

We shall denote the sequence of complex numbers resulting from the m^{th} stage of computation as $X_m(l)$, where $l=0,1,\ldots,N-1$, and $m=1,2,\ldots$, forming an input to the $(m+1)^{st}$ stage and producing an output $X_{m+1}(l)$ as the output from the $(m+1)^{st}$ stage of computations, it can be seen that the basic computation in flow graph of Figure

(2.3)

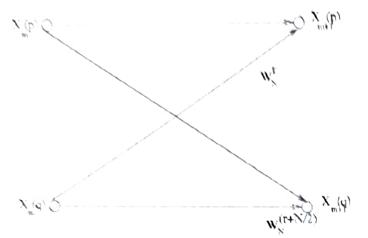


Figure 2.5: Flow graph of basic butterfly computation in Figure (2.3)

The equations represented by this flow graph are

$$X_{m+1}(p) = X_m(p) + W_N^r X_m(q)$$

$$X_{m+1}(q) = X_m(p) + W_N^{r+\frac{N}{2}} X_m(q) \qquad (2.8)$$

Because of the appearance of the flow graph of Figure (2.5), this computation is referred as a butterfly computation.

Equations (2.8) suggest a means of reducing the number of complex multiplications by a factor of 2. To see this we note that

$$W_N^{\frac{N}{2}} = e^{-j\frac{2\pi}{N}\cdot\frac{N}{2}} = e^{-j\pi} = -1$$

So the equations (2.8) become

$$X_{m+1}(p) = X_m(p) + W_N^r X_m(q)$$

$$X_{m+1}(q) = X_m(p) - W_N^r X_m(q) \qquad (2.9)$$

Equations (2.9) are represented in the flow graph of Figure (2.6).

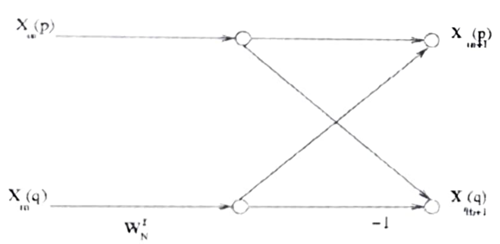


Figure 2.6: Flow graph of simplified butterfly computation requiring only one complex multiplication

Combining the observations in Figures (2.6), (2.5), (2.4) and (2.3), the efficient FFT algorithm in the computational flow graph representation for N=8 is obtained as shown in Figure (2.7). The algorithm requires $N/2\log_2 N$ complex multiplications and $N\log_2$

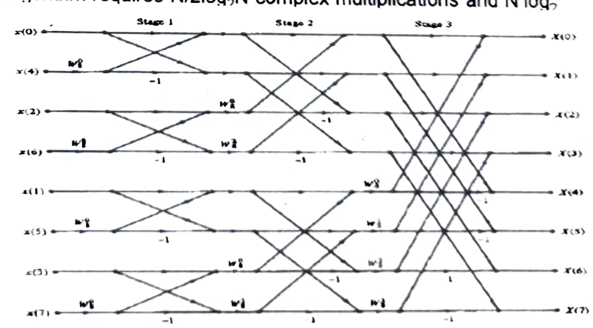


Figure 2.7: Complete flow graph of butterfly computation to compute 8-point DFT

Decimation-in-Frequency FFT Algorithm

- The decimation-in-time FFT algorithms were all based upon the decomposition of the DFT computation by forming smaller and smaller subsequences.
- Alternatively decimation-in-frequency FFT algorithms are all based upon decomposition of the DFT computation over X(k). For N a power of 2 i.e.

$$N = 2^{\nu}$$

we divide the input sequence into first half and the last half of points so that

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} x(n)W_N^{nk} + \sum_{\frac{N}{2}}^{N-1} x(n)W_N^{nk}, \qquad k = 0, 1, \ldots, N-1$$

or

$$X(k) = \sum_{n=0}^{\frac{N}{2}-1} x(n)W_N^{nk} + W_N^{\frac{N}{2}k} \sum_{n=0}^{\frac{N}{2}-1} x(n+\frac{N}{2})W_N^{nk}$$

$$= \sum_{n=0}^{\frac{N}{2}-1} [x(n) + (-1)^k x(n+\frac{N}{2})]W_N^{nk} \qquad \text{since, } W_N^{\frac{N}{2}k} = (-1)^k \quad (2.10)$$

- Separating k-even and k-odd, i.e. k=2r and k=2r+1, representing the even-numbered points and the odd-numbered points, respectively, so that

$$X(2r) = \sum_{n=0}^{\frac{N}{2}-1} [x(n) + x(n+\frac{N}{2})]W_{\frac{N}{2}}^{nr}, \qquad r = 0, 1, \ldots, \frac{N}{2}-1 \quad (2.11)$$

$$X(2r+1) = \sum_{n=0}^{\frac{N}{2}-1} [x(n) - x(n+\frac{N}{2})]W_N^{n}W_{\frac{N}{2}}^{nr}, \qquad r = 0, 1, \ldots, \frac{N}{2}-1 \quad (2.12)$$

Thus on the basis of Equations (2.11) and (2.12) with

$$g(n) = x(n) + x(n+\frac{N}{2})$$

and

$$h(n) = x(n) - x(n+\frac{N}{2})$$

The DFT can be computed by first forming the sequences g(n) and h(n), then computing $h(n)W_N^n$, and finally computing the N/2-point DFTs of these two sequences to obtain the even-numbered output points and odd-numbered output points, respectively.

The procedure suggested by Eqs. (2.11) and (2.12) is illustrated through signal flow graph for the case of 8-point DFT in Figure (2.8).

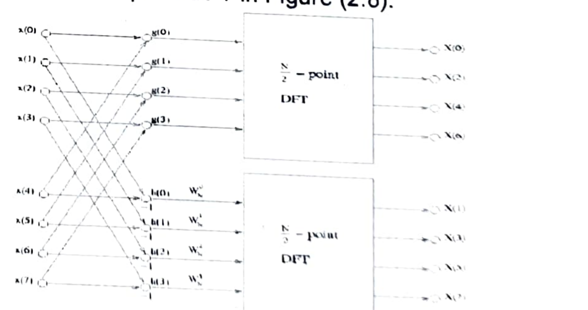


Figure 2.8: Flow graph of the decimation-in-frequency decomposition of an N-point computation into two $\frac{N}{2}$-point DFT computations, for $N = 8$

Proceeding in a manner similar to that followed in deriving the decimation-in-time algorithm, the final signal flow graph for computation is shown in Figure (2.9).

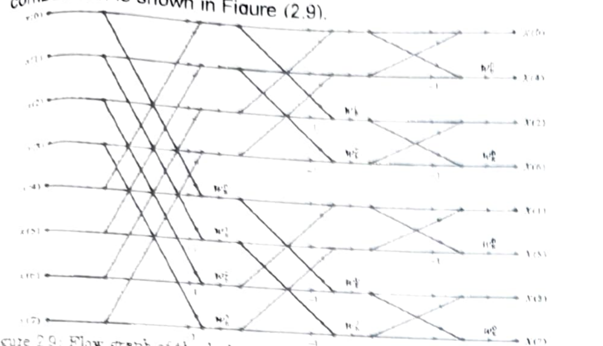


Figure 2.9 Flow graph of the decimation-in-frequency decomposition of an 8-point DFT computation

- By counting the arithmetic operations in Figure (2.9), and generalizing, we see that the computation of Figure (2.9) requires $N/2\log_2 N$ complex multiplications and $N\log_2 N$ complex additions. Thus the total computation is the same for decimation-in-frequency and decimation-in-time algorithms.
- Similar to decimation-in-time algorithm the computational flow graph shown in Figure (2.9) will indicate the in-place computation capability of decimation-in-frequency algorithm.
- Figure (2.9) is the transpose of Figure (2.7).

Decimation-In-Time FFT Algorithms

- Makes use of both symmetry and periodicity
- Consider special case of N an integer power of 2
- Separate x[n] into two sequence of length N/2
  - Even indexed samples in the first sequence
  - Odd indexed samples in the other sequence

$$X[k] = \sum_{n=0}^{N-1} x[n]e^{-j2\pi/Nkn} = \sum_{n\ \mathrm{even}}^{N-1} x[n]e^{-j2\pi/Nkn} + \sum_{n\ \mathrm{odd}}^{N-1} x[n]e^{-j2\pi/Nkn}$$

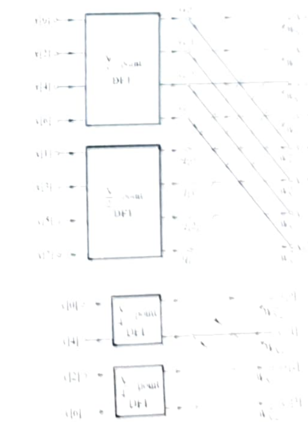
- Substitute variables n=2r for n even and n=2r+1 for odd

$$\begin{aligned} X[k] &= \sum_{r=0}^{N/2-1} x[2r]W_N^{2rk} + \sum_{r=0}^{N/2-1} x[2r+1]W_N^{(2r+1)k} \\ &= \sum_{r=0}^{N/2-1} x[2r]W_{N/2}^{rk} + W_N^k \sum_{r=0}^{N/2-1} x[2r+1]W_{N/2}^{rk} \\ &= G[k] + W_N^k H[k] \end{aligned}$$

- G[k] and H[k] are the N/2-point DFT's of each subsequence

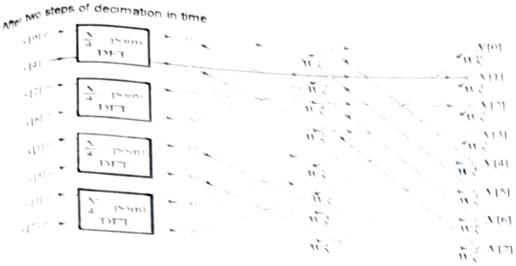
Decimation In Time

- 8-point DFT example using decimation-in-time
- Two N/2-point DFTs
  - $2(N/2)^2$ complex multiplications
  - $2(N/2)^2$ complex additions
- Combining the DFT outputs
  - N complex multiplications
  - N complex additions
- Total complexity
  - $N^2/2+N$ complex multiplications
  - $N^2/2+N$ complex additions
  - More efficient than direct DFT
- Repeat same process
  - Divide N/2-point DFTs into
  - Two N/4-point DFTs
  - Combine outputs

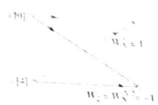


Decimation In Time Cont'd

- After two steps of decimation in time

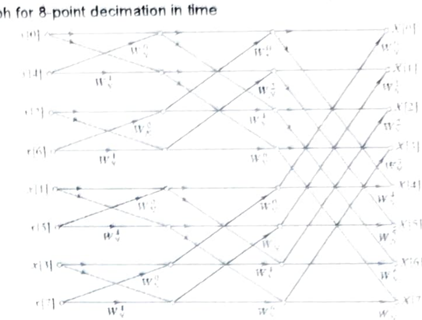


- Repeat until we're left with two-point DFT's



Decimation-In-Time FFT Algorithm

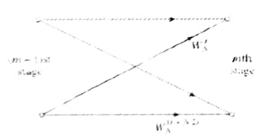
- Final flow graph for 8-point decimation in time



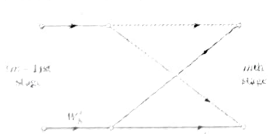
- Complexity:
  - $N\log_2 N$ complex multiplications and additions

Butterfly Computation

- Flow graph constitutes of butterflies



- We can implement each butterfly with one multiplication



- Final complexity for decimation-in-time FFT
  - $(N/2)\log_2 N$ complex multiplications and additions

In-Place Computation

- Decimation-in-time flow graphs require two sets of registers
  - Input and output for each stage
- Note the arrangement of the input indices
  - Bit reversed indexing

$$X_0[0] = x[0] \leftrightarrow X_0[000] = x[000]$$
$$X_0[1] = x[4] \leftrightarrow X_0[001] = x[100]$$
$$X_0[2] = x[2] \leftrightarrow X_0[010] = x[010]$$
$$X_0[3] = x[6] \leftrightarrow X_0[011] = x[110]$$
$$X_0[4] = x[1] \leftrightarrow X_0[100] = x[001]$$
$$X_0[5] = x[5] \leftrightarrow X_0[101] = x[101]$$
$$X_0[6] = x[3] \leftrightarrow X_0[110] = x[011]$$
$$X_0[7] = x[7] \leftrightarrow X_0[111] = x[111]$$

Decimation-In-Frequency FFT Algorithm

- The DFT equation

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk}$$

- Split the DFT equation into even and odd frequency indexes

$$X[2r] = \sum_{n=0}^{N-1} x[n] W_N^{n2r} = \sum_{n=0}^{N/2-1} x[n] W_N^{n2r} + \sum_{n=N/2}^{N-1} x[n] W_N^{n2r}$$

- Substitute variables to get

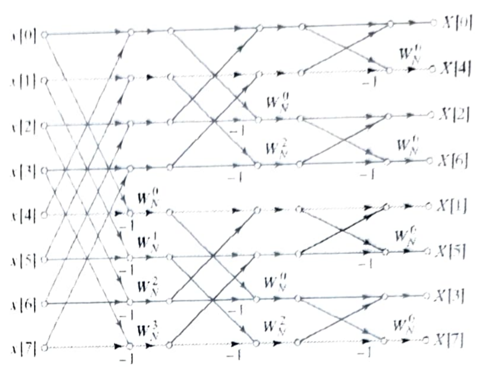
$$X[2r] = \sum_{n=0}^{N/2-1} x[n] W_N^{n2r} + \sum_{n=0}^{N/2-1} x[n+N/2] W_N^{(n+N/2)2r} = \sum_{n=0}^{N/2-1} (x[n] + x[n+N/2]) W_{N/2}^{nr}$$

- Similarly for odd-numbered frequencies

$$X[2r+1] = \sum_{n=0}^{N/2-1} (x[n] - x[n+N/2]) W_{N/2}^{n(2r+1)}$$

Decimation-In-Frequency FFT Algorithm

- Final flow graph for 8-point decimation in frequency



FFT vs. DFT

- The FFT is simply an algorithm for efficiently calculating the DFT
- Computational efficiency of an N-Point FFT:
  - DFT: N^2 Complex Multiplications
  - FFT: $(N/2) \log_2(N)$ Complex Multiplications

N	DFT Multiplications	FFT Multiplications	FFT Efficiency
256	65,536	1,024	64 : 1
512	262,144	2,304	114 : 1
1,024	1,048,576	5,120	205 : 1
2,048	4,194,304	11,264	372 : 1
4,096	16,777,216	24,576	683 : 1

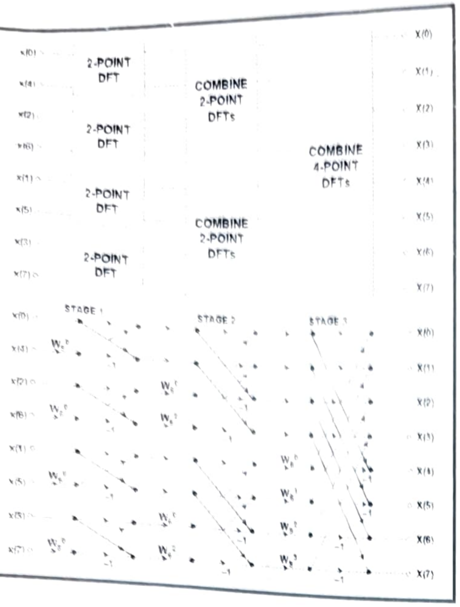
Bit Reversal

Decimal Number :	0	1	2	3	4	5	6	7
Binary Equivalent :	000	001	010	011	100	101	110	111
Bit-Reversed Binary :	000	100	010	110	001	101	011	111
Decimal Equivalent :	0	4	2	6	1	5	3	7

- The *bit reversal* algorithm used to perform the re-ordering of signals.
- The decimal index, n, is converted to its binary equivalent.
- The binary bits are then placed in reverse order, and converted back to a decimal number.
- Bit reversing is often performed in DSP hardware in the data address generator (DAG).

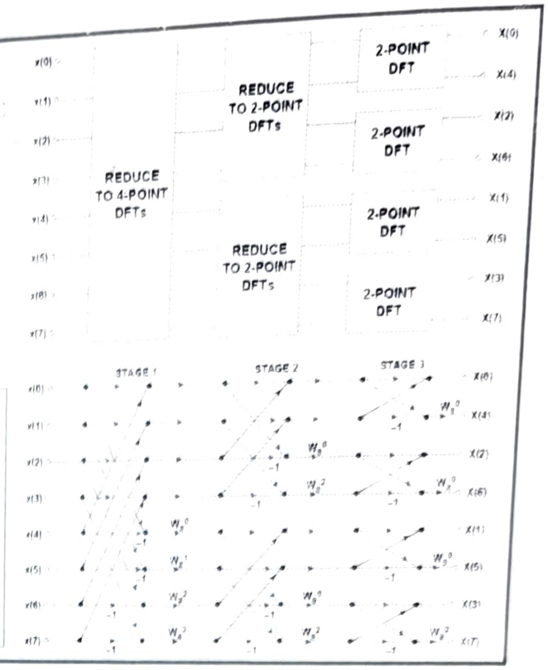
- Input signal must be properly re-ordered using a *bit reversal* algorithm
- In-place computation
- Number of stages: $\log_2 N$
- Stage 1: all the twiddle factors are 1
- Last Stage: the twiddle factors are in sequential order

	Stage 1	Stage 2	Stage 3	Stage $\log_2 N$
[illegible]	N/2	N/4	N/8	1
[illegible]	1	2	4	N/2
[illegible]	1	2	4	N/2
Twiddle Factor Exponents	(N/2)k, k=0	(N/4)k, k=0,1	(N/8)k, k=0,1,2,3	k, k=0 to N/2−1

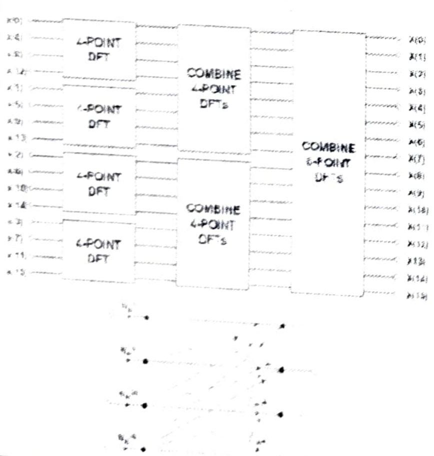


- Output signal must be properly re-ordered using a *bit reversal* algorithm
- In-place computation
- Number of stages: $\log_2 N$
- Stage 1: the twiddle factors are in sequential order
- Last Stage: all the twiddle factors are 1

	Stage 1	Stage 2	Stage 3	Stage $\log_2 N$
Number of Groups	1	2	4	N/2
Butterflies per Group	N/2	N/4	N/8	1
Dual-Node Spacing	N/2	N/4	N/8	1
Twiddle Factor Exponents	n, n=0 to N/2 - 1	2n, n=0 to N/4 - 1	4n, n=0 to N/8 - 1	(N/2)n, n=0



Radix-4 Decimation-In-Time FFT Algorithm



- A radix-4 FFT combines two stages of a radix-2 FFT into one, so that half as many stages are required.
- The radix-4 butterfly is consequently larger and more complicated than a radix-2 butterfly.
- N/4 butterflies are used in each of $(\log_2 N)/2$ stages, which is one quarter the number of butterflies in a radix-2 FFT.
- Addressing of data and twiddle factors is more complex, a radix-4 FFT requires fewer calculations than a radix-2 FFT.
- It can compute a radix-4 FFT significantly faster than a radix-2 FFT

Inverse Discrete Fourier Transform (IDFT)

The inverse discrete Fourier transform (IDFT) is given by

$$x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X(k) W_N^{-kn}, \qquad n = 0, 1, \ldots, N-1 \qquad (2.13)$$

which is structurally similar to DFT,

$$X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}, \qquad k = 0, 1, \ldots, N-1 \qquad (2.14)$$

The change we notice is in the multiplication factor 1/N} and replacement of W_N^{kn} by W_N^{-kn}, and the interchange of signals x(n) and X(k) in the expressions and the index for summation.

- Thus in Figure (2.7) and (2.9), if we exercise the above changes, the changed signal flow graphs will become algorithms for IDFT and referred as IFFT algorithms.

Example

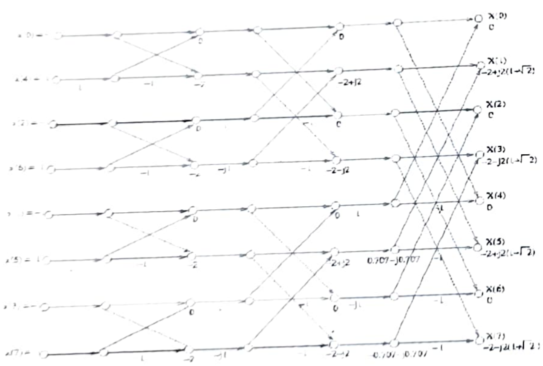
- Using decimation-in-time FFT algorithm compute DFT of the sequence

  {-1 −1 −1 −1 1 1 1 1}

- Solution: Twiddle factors are

$$W_8 = e^{-j\frac{2\pi}{8}} = e^{-j\frac{\pi}{4}} = 0.707 - j0.707$$
$$W_8^0 = 1$$
$$W_8^1 = 0.707 - j0.707$$
$$W_8^2 = -j1$$
$$W_8^3 = -0.707 - j0.707$$

Solution and signal flow graph of the example



Thank You

SHRI. B. V. V. SANGHA'S

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOTE – 587 102.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING.

B. E. VII – SEMESTER MULTIMEDIA COMMUNICATIONS ASSIGNMENT – I.

Sl.No.	Day	Date	U. S. No.	Time
1.	Monday	04/11/2024	2BA21EC001, 2BA21EC002, 2BA21EC005, 2BA21EC006, 2BA21EC008, 2BA21EC009, 2BA21EC011, 2BA21EC013, 2BA21EC015, 2BA21EC016, 2BA21EC017, 2BA21EC020, 2BA21EC021, 2BA21EC022, 2BA21EC023, 2BA21EC024, 2BA21EC025, 2BA21EC028, 2BA21EC031, 2BA21EC032, 2BA21EC035, 2BA21EC037, 2BA21EC038.	0200 PM to 04:00 PM
2.	Tuesday	05/11/2024	2BA21EC040, 2BA21EC043, 2BA21EC045, 2BA21EC048, 2BA21EC049, 2BA21EC050, 2BA21EC054, 2BA21EC055, 2BA21EC057, 2BA21EC058, 2BA21EC060, 2BA21EC064, 2BA21EC065, 2BA21EC066, 2BA21EC069, 2BA21EC073, 2BA21EC075, 2BA21EC078, 2BA21EC079, 2BA21EC082, 2BA21EC085, 2BA21EC087, 2BA21EC088.	0200 PM to 04:00 PM
3.	Wednsday	06/11/2024	2BA21EC090, 2BA21EC093, 2BA21EC096, 2BA21EC081, 2BA21EC083, 2BA21EC084, 2BA21EC086, 2BA21EC089, 2BA21EC091, 2BA21EC092, 2BA21EC094, 2BA21EC095, 2BA21EC097, 2BA21EC098, 2BA21EC099, 2BA21EC102, 2BA21EC108, 2BA21EC110, 2BA21EC111, 2BA21EC112, 2BA21EC113, 2BA21EC115.	0200 PM to 04:00 PM
4.	Thursday	07/11/2024	2BA21EC100, 2BA21EC101, 2BA21EC103, 2BA21EC104, 2BA21EC105, 2BA21EC106, 2BA21EC107, 2BA21EC109, 2BA21EC114, 2BA21EC117, 2BA21EC119, 2BA21EC121, 2BA21EC124, 2BA21EC125, 2BA21EC077, 2BA21EC003, 2BA21EC004, 2BA21EC007, 2BA21EC012, 2BA21EC014, 2BA21EC018, 2BA21EC019, 2BA21EC026.	0200 PM to 04:00 PM
5.	Friday	08/11/2024	2BA21EC027, 2BA21EC029, 2BA21EC030, 2BA21EC033, 2BA21EC034, 2BA21EC036, 2BA21EC039, 2BA21EC042, 2BA21EC044, 2BA21EC046, 2BA21EC047, 2BA21EC051, 2BA21EC052, 2BA21EC053, 2BA21EC056, 2BA21EC059, 2BA21EC061, 2BA21EC062, 2BA21EC063, 2BA21EC067, 2BA21EC068, 2BA21EC070, 2BA21EC071.	0200 PM to 04:00 PM
6.	Saturday	09/11/2024	2BA21EC072, 2BA21EC074, 2BA21EC076, 2BA21EC080, 2BA21EC118, 2BA21EC120, 2BA21EC123, 2BA21EC126, 2BA21EC127, 2BA21EC128, 2BA21EC129, 2BA22EC404, 2BA22EC402, 2BA22EC400, 2BA22EC401, 2BA22EC405, 2BA22EC406, 2BA22EC403, 2BA22EC409, 2BA22EC407, 2BA22EC410, 2BA22EC408, 2BA22EC411.	12:15 PM To 02:15 PM

Jmallapur 26/10/

H.O.D

Professor and Head
Department of Electronics & Communication Engg.
Basaveshwar Engineering College.
BAGALKOT - 587 102

SHRI. B. V. V. SANGHA'S

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOTE – 587 102.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING.

. E. VII – SEMESTER MULTIMEDIA COMMUNICATIONS ASSIGNMENT – II.

.No.	Day	Date	U. S. No.	Time
.	Monday	11/11/2024	2BA21EC001, 2BA21EC002, 2BA21EC005, 2BA21EC006, 2BA21EC008, 2BA21EC009, 2BA21EC011, 2BA21EC013, 2BA21EC015, 2BA21EC016, 2BA21EC017, 2BA21EC020, 2BA21EC021, 2BA21EC022, 2BA21EC023, 2BA21EC024, 2BA21EC025, 2BA21EC028, 2BA21EC031, 2BA21EC032, 2BA21EC035, 2BA21EC037, 2BA21EC038.	0200 PM to 04:00 PM
2.	Tuesday	12/11/2024	2BA21EC040, 2BA21EC043, 2BA21EC045, 2BA21EC048, 2BA21EC049, 2BA21EC050, 2BA21EC054, 2BA21EC055, 2BA21EC057, 2BA21EC058, 2BA21EC060, 2BA21EC064, 2BA21EC065, 2BA21EC066, 2BA21EC069, 2BA21EC073, 2BA21EC075, 2BA21EC078, 2BA21EC079, 2BA21EC082, 2BA21EC085, 2BA21EC087, 2BA21EC088.	0200 PM to 04:00 PM
3.	Wednsday	13/11/2024	2BA21EC090, 2BA21EC093, 2BA21EC096, 2BA21EC081, 2BA21EC083, 2BA21EC084, 2BA21EC086, 2BA21EC089, 2BA21EC091, 2BA21EC092, 2BA21EC094, 2BA21EC095, 2BA21EC097, 2BA21EC098, 2BA21EC099, 2BA21EC102, 2BA21EC108, 2BA21EC110, 2BA21EC111, 2BA21EC112, 2BA21EC113, 2BA21EC115.	0200 PM to 04:00 PM
4.	Thursday	14/11/2024	2BA21EC100, 2BA21EC101, 2BA21EC103, 2BA21EC104, 2BA21EC105, 2BA21EC106, 2BA21EC107, 2BA21EC109, 2BA21EC114, 2BA21EC117, 2BA21EC119, 2BA21EC121, 2BA21EC124, 2BA21EC125, 2BA21EC077, 2BA21EC003, 2BA21EC004, 2BA21EC007, 2BA21EC012, 2BA21EC014, 2BA21EC018, 2BA21EC019, 2BA21EC026.	0200 PM to 04:00 PM
5.	Friday	15/11/2024	2BA21EC027, 2BA21EC029, 2BA21EC030, 2BA21EC033, 2BA21EC034, 2BA21EC036, 2BA21EC039, 2BA21EC042, 2BA21EC044, 2BA21EC046, 2BA21EC047, 2BA21EC051, 2BA21EC052, 2BA21EC053, 2BA21EC056, 2BA21EC059, 2BA21EC061, 2BA21EC062, 2BA21EC063, 2BA21EC067, 2BA21EC068, 2BA21EC070, 2BA21EC071.	0200 PM to 04:00 PM
6.	Saturday	16/11/2024	2BA21EC072, 2BA21EC074, 2BA21EC076, 2BA21EC080, 2BA21EC118, 2BA21EC120, 2BA21EC123, 2BA21EC126, 2BA21EC127, 2BA21EC128, 2BA21EC129, 2BA22EC404, 2BA22EC402, 2BA22EC400, 2BA22EC401, 2BA22EC405, 2BA22EC406, 2BA22EC403, 2BA22EC409, 2BA22EC407, 2BA22EC410, 2BA22EC408, 2BA22EC411.	12:15 PM To 02:15 PM

Jmallappa 26/10/24

H.O.D

Professor and Head
Department of Electronics & Communication Engg
Basaveshwar Engineering College,
BAGALKOT - 587 102

SHRI. B. V. V. SANGHA'S

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOTE – 587 102.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING.

B. E. VII – SEMESTER MULTIMEDIA COMMUNICATIONS ASSIGNMENT – I.

Sl.No.	Day	Date	U. S. No.	Time
1.	Monday	04/11/2024	2BA21EC001, 2BA21EC002, 2BA21EC005, 2BA21EC006, 2BA21EC008, 2BA21EC009, 2BA21EC011, 2BA21EC013, 2BA21EC015, 2BA21EC016, 2BA21EC017, 2BA21EC020, 2BA21EC021, 2BA21EC022, 2BA21EC023, 2BA21EC024, 2BA21EC025, 2BA21EC028, 2BA21EC031, 2BA21EC032, 2BA21EC035, 2BA21EC037, 2BA21EC038.	0200 PM to 04:00 PM
2.	Tuesday	05/11/2024	2BA21EC040, 2BA21EC043, 2BA21EC045, 2BA21EC048, 2BA21EC049, 2BA21EC050, 2BA21EC054, 2BA21EC055, 2BA21EC057, 2BA21EC058, 2BA21EC060, 2BA21EC064, 2BA21EC065, 2BA21EC066, 2BA21EC069, 2BA21EC073, 2BA21EC075, 2BA21EC078, 2BA21EC079, 2BA21EC082, 2BA21EC085, 2BA21EC087, 2BA21EC088.	0200 PM to 04:00 PM
3.	Wednsday	06/11/2024	2BA21EC090, 2BA21EC093, 2BA21EC096, 2BA21EC081, 2BA21EC083, 2BA21EC084, 2BA21EC086, 2BA21EC089, 2BA21EC091, 2BA21EC092, 2BA21EC094, 2BA21EC095, 2BA21EC097, 2BA21EC098, 2BA21EC099, 2BA21EC102, 2BA21EC108, 2BA21EC110, 2BA21EC111, 2BA21EC112, 2BA21EC113, 2BA21EC115.	0200 PM to 04:00 PM
4.	Thursday	07/11/2024	2BA21EC100, 2BA21EC101, 2BA21EC103, 2BA21EC104, 2BA21EC105, 2BA21EC106, 2BA21EC107, 2BA21EC109, 2BA21EC114, 2BA21EC117, 2BA21EC119, 2BA21EC121, 2BA21EC124, 2BA21EC125, 2BA21EC077, 2BA21EC003, 2BA21EC004, 2BA21EC007, 2BA21EC012, 2BA21EC014, 2BA21EC018, 2BA21EC019, 2BA21EC026.	0200 PM to 04:00 PM
5.	Friday	08/11/2024	2BA21EC027, 2BA21EC029, 2BA21EC030, 2BA21EC033, 2BA21EC034, 2BA21EC036, 2BA21EC039, 2BA21EC042, 2BA21EC044, 2BA21EC046, 2BA21EC047, 2BA21EC051, 2BA21EC052, 2BA21EC053, 2BA21EC056, 2BA21EC059, 2BA21EC061, 2BA21EC062, 2BA21EC063, 2BA21EC067, 2BA21EC068, 2BA21EC070, 2BA21EC071.	0200 PM to 04:00 PM
6.	Saturday	09/11/2024	2BA21EC072, 2BA21EC074, 2BA21EC076, 2BA21EC080, 2BA21EC118, 2BA21EC120, 2BA21EC123, 2BA21EC126, 2BA21EC127, 2BA21EC128, 2BA21EC129, 2BA22EC404, 2BA22EC402, 2BA22EC400, 2BA22EC401, 2BA22EC405, 2BA22EC406, 2BA22EC403, 2BA22EC409, 2BA22EC407, 2BA22EC410, 2BA22EC408, 2BA22EC411.	12:15 PM To 02:15 PM

Jmallayya 26/10/24

Professor and Head
Department of Electronics & Communication Engg.
Basaveshwar Engineering College

H. O. D

SHRI. B. V. V. SANGHA'S

BASAVESHWAR ENGINEERING COLLEGE, BAGALKOTE – 587 102.

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING.

B. E. VII – SEMESTER MULTIMEDIA COMMUNICATIONS ASSIGNMENT – II.

Sl.No.	Day	Date	U. S. No.	Time
1.	Monday	11/11/2024	2BA21EC001, 2BA21EC002, 2BA21EC005, 2BA21EC006, 2BA21EC008, 2BA21EC009, 2BA21EC011, 2BA21EC013, 2BA21EC015, 2BA21EC016, 2BA21EC017, 2BA21EC020, 2BA21EC021, 2BA21EC022, 2BA21EC023, 2BA21EC024, 2BA21EC025, 2BA21EC028, 2BA21EC031, 2BA21EC032, 2BA21EC035, 2BA21EC037, 2BA21EC038.	0200 PM to 04:00 PM
2.	Tuesday	12/11/2024	2BA21EC040, 2BA21EC043, 2BA21EC045, 2BA21EC048, 2BA21EC049, 2BA21EC050, 2BA21EC054, 2BA21EC055, 2BA21EC057, 2BA21EC058, 2BA21EC060, 2BA21EC064, 2BA21EC065, 2BA21EC066, 2BA21EC069, 2BA21EC073, 2BA21EC075, 2BA21EC078, 2BA21EC079, 2BA21EC082, 2BA21EC085, 2BA21EC087, 2BA21EC088.	0200 PM to 04:00 PM
3.	Wednsday	13/11/2024	2BA21EC090, 2BA21EC093, 2BA21EC096, 2BA21EC081, 2BA21EC083, 2BA21EC084, 2BA21EC086, 2BA21EC089, 2BA21EC091, 2BA21EC092, 2BA21EC094, 2BA21EC095, 2BA21EC097, 2BA21EC098, 2BA21EC099, 2BA21EC102, 2BA21EC108, 2BA21EC110, 2BA21EC111, 2BA21EC112, 2BA21EC113, 2BA21EC115.	0200 PM to 04:00 PM
4.	Thursday	14/11/2024	2BA21EC100, 2BA21EC101, 2BA21EC103, 2BA21EC104, 2BA21EC105, 2BA21EC106, 2BA21EC107, 2BA21EC109, 2BA21EC114, 2BA21EC117, 2BA21EC119, 2BA21EC121, 2BA21EC124, 2BA21EC125, 2BA21EC077, 2BA21EC003, 2BA21EC004, 2BA21EC007, 2BA21EC012, 2BA21EC014, 2BA21EC018, 2BA21EC019, 2BA21EC026.	0200 PM to 04:00 PM
5.	Friday	15/11/2024	2BA21EC027, 2BA21EC029, 2BA21EC030, 2BA21EC033, 2BA21EC034, 2BA21EC036, 2BA21EC039, 2BA21EC042, 2BA21EC044, 2BA21EC046, 2BA21EC047, 2BA21EC051, 2BA21EC052, 2BA21EC053, 2BA21EC056, 2BA21EC059, 2BA21EC061, 2BA21EC062, 2BA21EC063, 2BA21EC067, 2BA21EC068, 2BA21EC070, 2BA21EC071.	0200 PM to 04:00 PM
6.	Saturday	16/11/2024	2BA21EC072, 2BA21EC074, 2BA21EC076, 2BA21EC080, 2BA21EC118, 2BA21EC120, 2BA21EC123, 2BA21EC126, 2BA21EC127, 2BA21EC128, 2BA21EC129, 2BA22EC404, 2BA22EC402, 2BA22EC400, 2BA22EC401, 2BA22EC405, 2BA22EC406, 2BA22EC403, 2BA22EC409, 2BA22EC407, 2BA22EC410, 2BA22EC408, 2BA22EC411.	12:15 PM To 02:15 PM

Mallapur 26/10/24

Professor and Head
Department of Electronics & Communication Engg.
Basaveshwar Engineering College,
BAGALKOT

H. O. D

IoT APPLICATIONS

MCQ ON IOT

Sign in to Google to save your progress. Learn more

* Indicates required question

What is the Internet of Things (IoT)? * 1 point

- a) A system that connects all types of devices to the internet
- b) A technology that allows for remote control of home appliances
- c) A tool for managing computer networks
- d) A platform for online shopping

What is the main purpose of IoT? * 1 point

- a) To collect and analyze data from connected devices
- b) To control home appliances remotely
- c) To improve online shopping experiences
- d) To create a virtual reality environment

. Which of the following is an example of an IoT device? * 1 point

- a) A laptop computer
- b) A fitness tracker
- c) A television set
- d) A landline telephone

Which wireless communication protocol is commonly used in IoT devices? * 1 point

- a) Wi-Fi
- b) Bluetooth
- c) NFC
- d) Infrared

What is a sensor in IoT? * 1 point

- a) A device that converts physical or environmental parameters into digital signals
- b) A device that connects to the internet and sends data
- c) A device that provides a graphical user interface
- d) A device that runs software programs

What is a gateway in IoT? * 1 point

- a) A device that connects IoT devices to the internet
- b) A device that stores data collected by IoT devices
- c) A device that analyzes data collected by IoT devices
- d) A device that provides a user interface for IoT devices

Which of the following is an example of an IoT application in healthcare? * 1 point

- a) Online shopping for medical supplies
- b) Remote patient monitoring
- c) Electronic medical record management
- d) Telemedicine consultations

1 point

What is a smart home in IoT? *

- a) A home that is equipped with IoT devices and systems
- b) A home that is powered by renewable energy sources
- c) A home that uses advanced security systems
- d) A home that is completely automated

Which of the following is an example of an IoT application in agriculture? * 1 point

- a) Online crop sales platform
- b) Farm management software
- c) Weather forecasting app
- d) Smart irrigation system

1 point

What is edge computing in IoT? *

- a) A process of analyzing data at the source of generation, rather than sending it to a centralized location
- b) A process of analyzing data on a remote server
- c) A process of storing data on a local device
- d) A process of transmitting data over a wired network

Which of the following is a disadvantage of IoT? * 1 point

- a) Increased efficiency and productivity
- b) Improved decision-making and analytics
- c) Privacy and security concerns
- d) Greater connectivity and collaboration

Which of the following is an example of an IoT application in transportation? * 1 point

- a) Online ticket booking system
- b) GPS navigation system
- c) Traffic management software
- d) Automotive diagnostic tool

Which of the following is an example of an IoT application in retail? * 1 point

- a) Online advertising platform
- b) Customer relationship management software
- c) Inventory management system
- d) Point of sale system

Which of the following is an example of an IoT application in energy management? * 1 point

- a) Online billing and payment platform
- b) Smart grid technology
- c) Energy efficiency audit software
- d) Renewable energy generation system

1 point

What is the role of cloud computing in IoT? *

- a) To store and process data collected by IoT devices
- b) To provide connectivity between IoT devices
- c) To analyze data generated by IoT devices
- d) To manage and control IoT devices

Untitled Question

- Option 1

Submit

Clear form

Department of Electronics and Communication Engg.

Innovation in Teaching Learning Process

Image Processing_UEC635N

A rubrics prepared for self evaluation of students for their presentation of assignment: Questioner and responses collected

Evaluation of Presentation Assignment 1(UEC632N)

Excellent (5), Good (4), Satisfactory (3), Needs improvement (2), Inadequate (1)

	5	4	3	2	1
Demonstrates a deep understanding of the topic.					
Shows a solid grasp of the material.					
Presents information accurately but lacks depth.					
Contains inaccuracies or incomplete information.					
Fails to convey a clear understanding of the topic.					

Organization and Structure *

Excellent (5), Good (4), Satisfactory (3), Needs improvement (2), Inadequate (1)

	5	4	3	2	1
Presentation is well-structured and logical.					
Information is presented in a clear sequence.					
Structure is somewhat confusing.					
Lack of organization; difficult to follow.					
Chaotic and disorganized; difficult to comprehend.					

8. Delivery and Engagement *

Excellent (5), Good (4), Satisfactory (3), Needs improvement (2), Inadequate (1)

Mark only one oval per row.

	5	4	3	2	1
Engages the audience with a confident demeanor.					
Maintains good eye contact and vocal clarity.					
Some engagement but lacks consistent enthusiasm.					
Appears nervous or disinterested.					
Demonstrates little effort to engage the audience.					

8. **Delivery and Engagement** *

Excellent (5), Good (4), Satisfactory (3), Needs Improvement (2),Inadequate (1)

Mark only one oval per row.

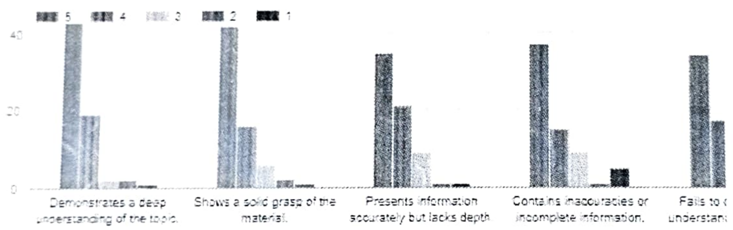
	5	4	3	2	1
Engages the audience with a confident demeanor.					
Maintains good eye contact and vocal clarity.					
Some engagement but lacks consistent enthusiasm.					
Appears nervous or disinterested.					
Demonstrates little effort to engage the audience.					

10. Time Management *

Excellent (5), Good (4), Satisfactory (3), Needs Improvement (2),Inadequate (1)

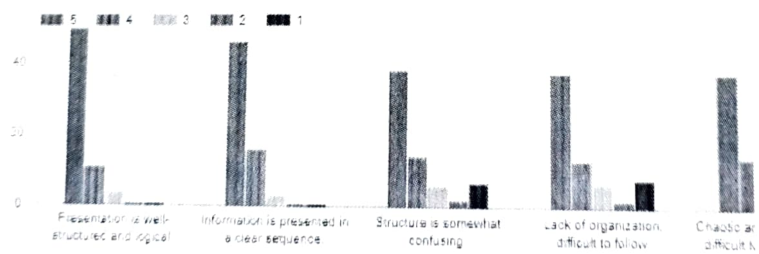
Mark only one oval per row.

	5	4	3	2	1
Stays within the allocated time limit.					
Finishes slightly before or after the time limit.					
Goes over or significantly under the time limit.					
Exceeds time limit without covering all points.					
Presentation is too short and lacks substance.					

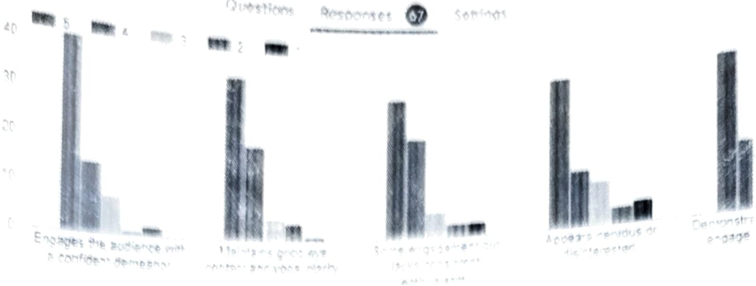


Organization and Structure

Copy chart

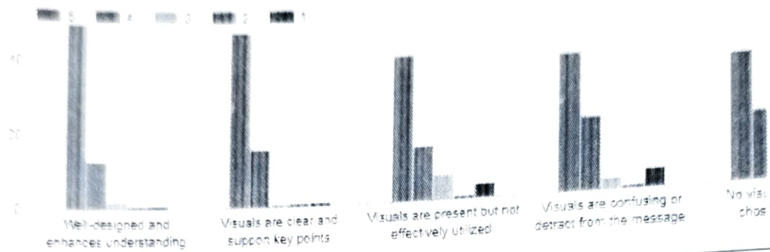


Questions Responses 67 Settings



Visual Aids

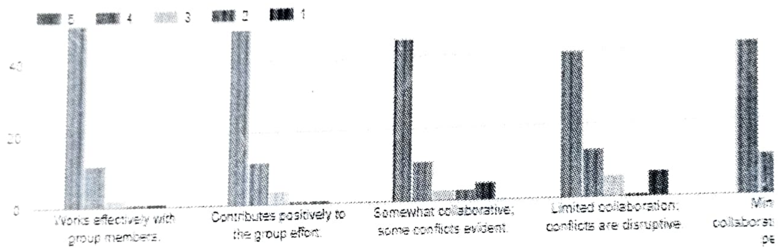
Copy chart



Questions Responses 67 Settings

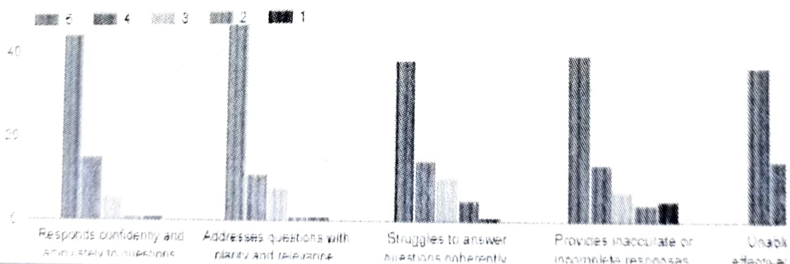
Copy chart

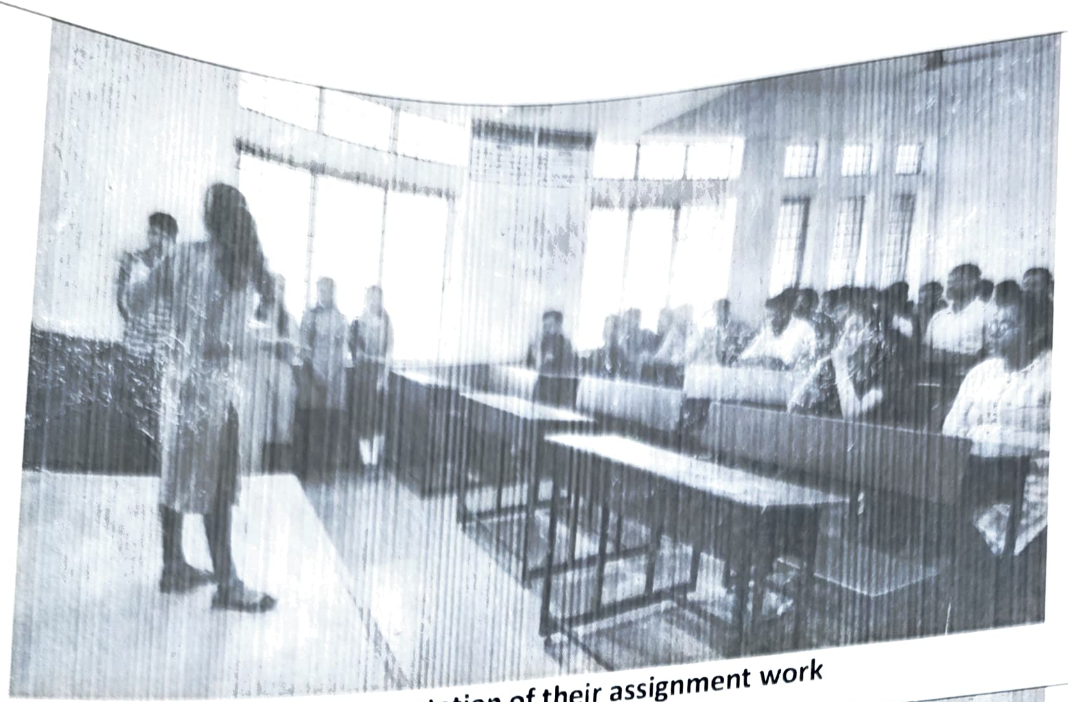
Collaboration with team members



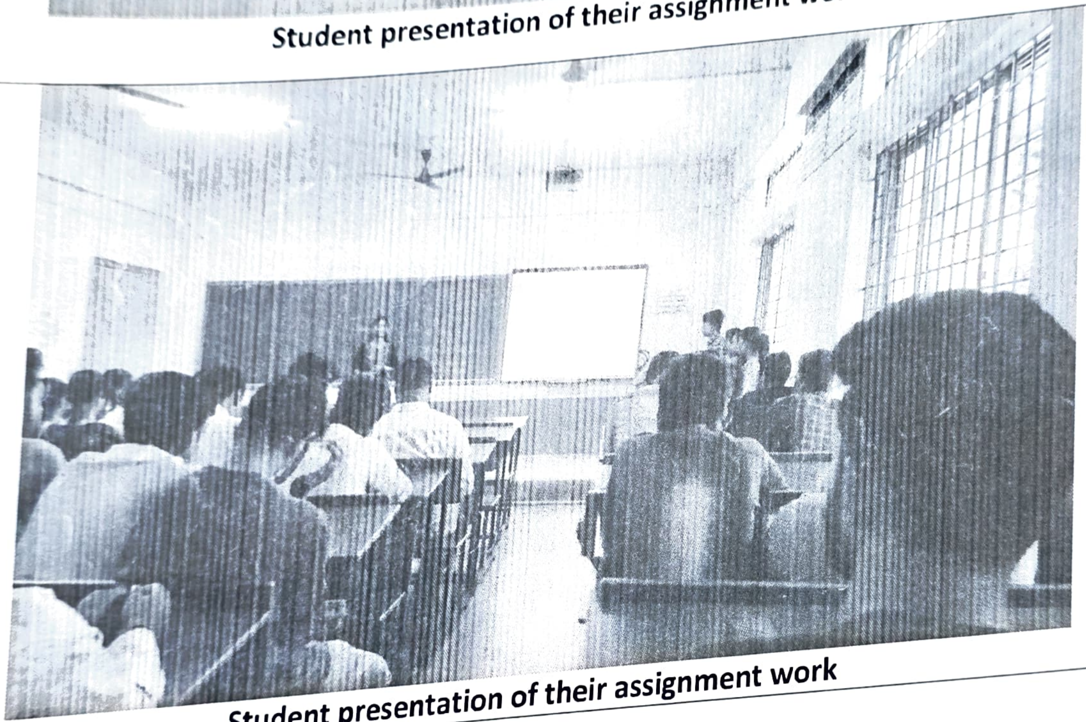
Q&A Session

Copy chart





Student presentation of their assignment work



Student presentation of their assignment work

(Dr. Vijaylakshmi. S.T)

Basaveshwar Engineering College, Bagalkot
Department of Electronics and Communication Engineering
(Academic Year 2023-24)
Course Project Details

Sub: Computer Networks
Div: A

Subject Code: 21UEC603C
Staff Incharge: Dr. Ashok Sutagundar

SL No.	USN	Name of student	Project title
1)	2BA21EC050	Panakaj Biradar	GLBP and Frame Relay Protocols
2)	2BA21EC054	Pooja P Patil	Hotel Management Network Design
3)	2BA21EC055	Pragati Pattanshetti	IOT based Smart Garden using CISCO Packet tracer
4)	2BA21EC057	Prajwal Desai	Fire Alarm System
5)	2BA21EC058	Prajwal H. K.	Temperature sensor data Acquisition
6)	2BA21EC060	Promod Hiremath	Simulation od Distance Vector Routing Protocol
7)	2BA21EC0064	Pratiksha Dodamani	Simulation of IPV4 and Enhance Gateway Routing Protocol
8)	2BA21EC0065	Praveen Hatti	Simulation of Domain System Protocol using Cisco Packet Tracer
9)	2BA21EC016	Anand Hiremani	LCP and PAGP protocols using CISCO Packet tracer
10)	2BA21EC017	A. M. Desai	Hot standby Routing Protocol and NAT protocol translation
11)	2BA21EC020	Arun Budni	Smart Hospital Environment
12)	2BA21EC021	Arundati Bhavikatti	Open VPN and Wire Guard protocol
13)	2BA21EC022	Ashiwini Chouvan `	Smart Agriculture using Cisco Packet tracer
14)	2BA21EC023	Avinash	Configuration of DNS server and Client Link Layer Discovery
15)	2BA21EC024	Basavaraj Kohalli	Controlling windows based on CO Level
16)	2BA21EC025	Basavaraj urf Koushik	Solar Powered IoT Devices using Cisco Packet Tracer
17)	2BA21EC029	Chandrashekar G. Hadalagi	Simulation of ICMP protocol using Cisco Packet Tracer
18)	2BA21EC031	Gagan Bhairamatti	Smart Day/Night using Cisco Packet Tracer
19)	2BA21EC032	Ganga Patil	Thief Catcher using IoT system
20)	2BA21EC035	Jyothi Alagodi	Hospital Managemnt System using Cisco Packet Tracer
21)	2BA21EC037	Karthik Kulkarni	Simple Office Networking
22)	2BA21EC038	Kiran Muradi	Smart Garden using IoT
23)	2BA21EC040	Laxmi Nandikolmath	IoT based solar panel
24)	2BA21EC043	Mohmad Rehan Pattankundi	Simulation of CDP and PAT protocol
25)	2BA21EC045	Narayan N Malabasari	Room Automation using Cisco Packet Tracer

26)	2BA21EC048	Nitish	Campus Network Management
27)	2BA21EC049	Pallavi Babu Halki	Smart Home Automation using Cisco Packet tracer

(Dr. Ashok Sutagundar)