Research Migration Project

https://esim.fossee.in/research-migration-project

The Research Migration Project is an initiative of FOSSEE, IIT Bombay that promotes the use of eSim for reproducing published research circuits originally implemented using proprietary simulation tools. The objective is to migrate these validated designs to eSim to build an open source resource database.

Name of the participant : Bellana venkata chaitanya

Affiliation / Institution : Department of *Electronics and communication engineering, Rajiv Gandhi University Of Knowledge and Technologies,Nuzvid,Vijayawada,India*

Title of the circuit : Design of Single Precision Floating Point Arithmetic logic Unit Using eSim

Theory/Description : Floating-point arithmetic is essential for applications requiring high precision numerical computations. The IEEE 754 standard defines the format for single precision (32-bit) floating point numbers, consisting of 1 sign bit, 8 exponent bits, and 23 mantissa bits. This work presents a complete floating point arithmetic logic unit (FP-ALU) capable of performing addition, subtraction, multiplication, and division operations.

Floating point numbers are used in many applications such as telecommunications, medical imaging, radar,etc.

In top-down design approach, four arithmetic modules,addition, subtraction, multiplication and division are combined to form a floating point ALU unit. Each module is independent to each other. In this paper, the implementation of a floating point ALU is designed and simulated. Here we presents the design of a single precision floating point arithmetic logic unit. The operations are performed on 32-bit operands. The algorithms of addition, subtraction, division and multiplication are modeled in Verilog HDL using XILINX VIVADOand an efficient algorithm for addition and subtraction module is developed in order to reduce the number of gates used.

Expected Outcome/outputs :

The addition, subtraction, multiplication, and division operations are performed based on the 4-bit opcode input.

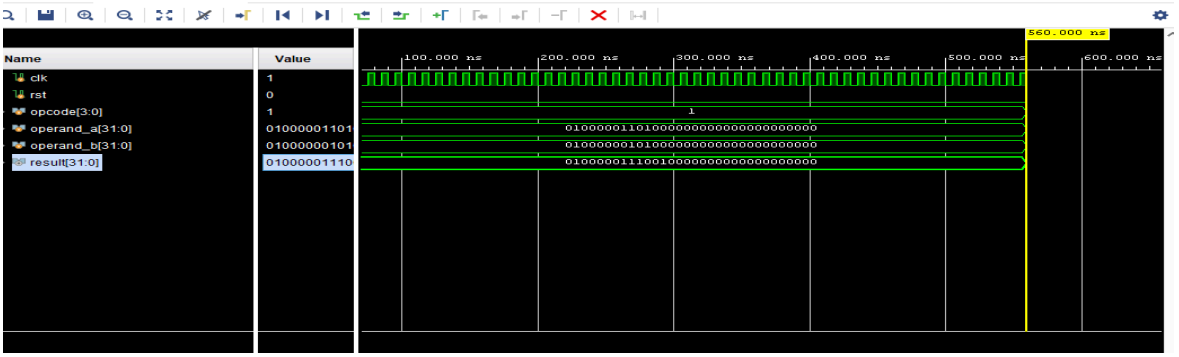
for all cases input1 is `01000001101000000000000000000000` which is 20.0 in decimal

input2 is `01000000101000000000000000000000`which is 5.0 in decimal

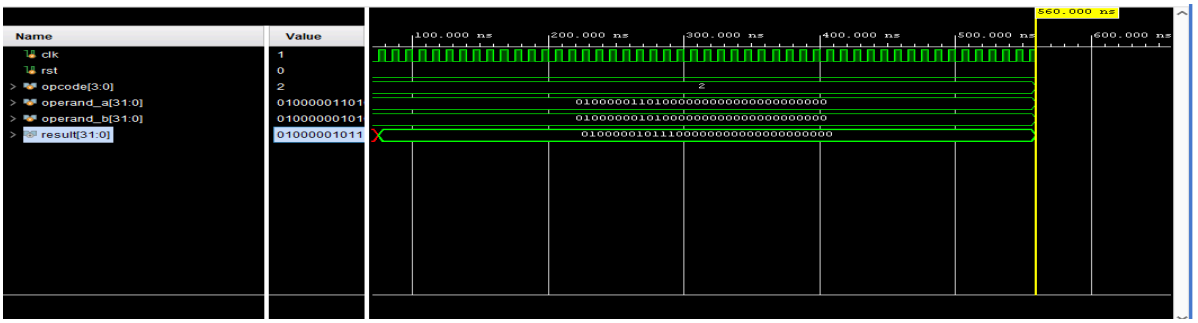
opcodes: Addition:0001 Multiplication:0011

Subtraction:0010 Division:0100

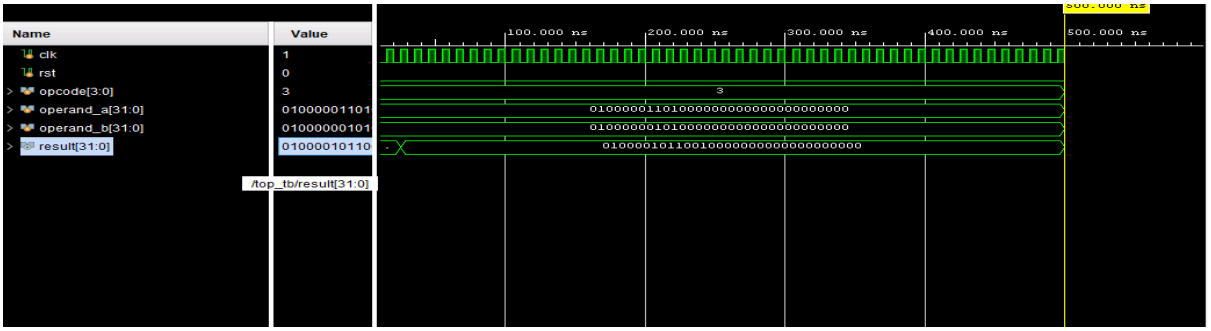
Addition: output obtained is `01000001110010000000000000000000`.which is 25.0 in decimal



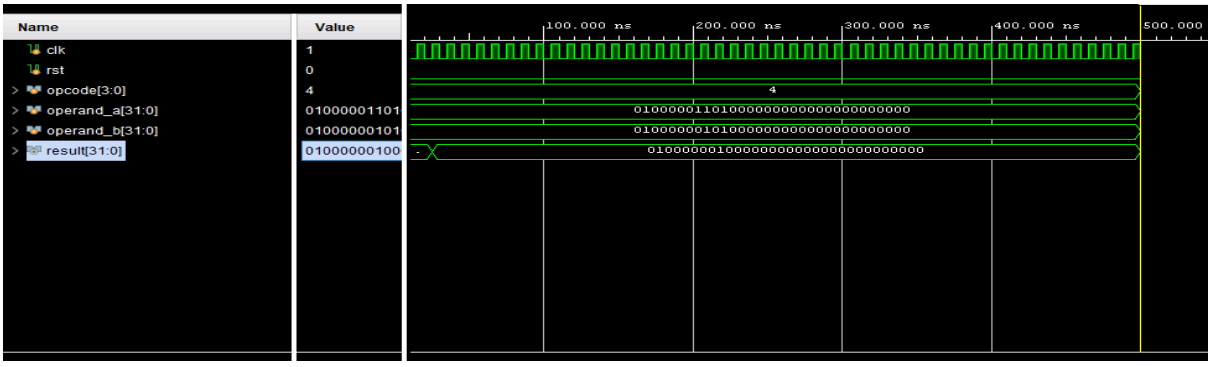
subtraction:the output obtained is `01000001011100000000000000000000`.which is 15.0 in decimal



multiplication:the output obtained is `01000001011001000000000000000000`.which is 100 in decimal.

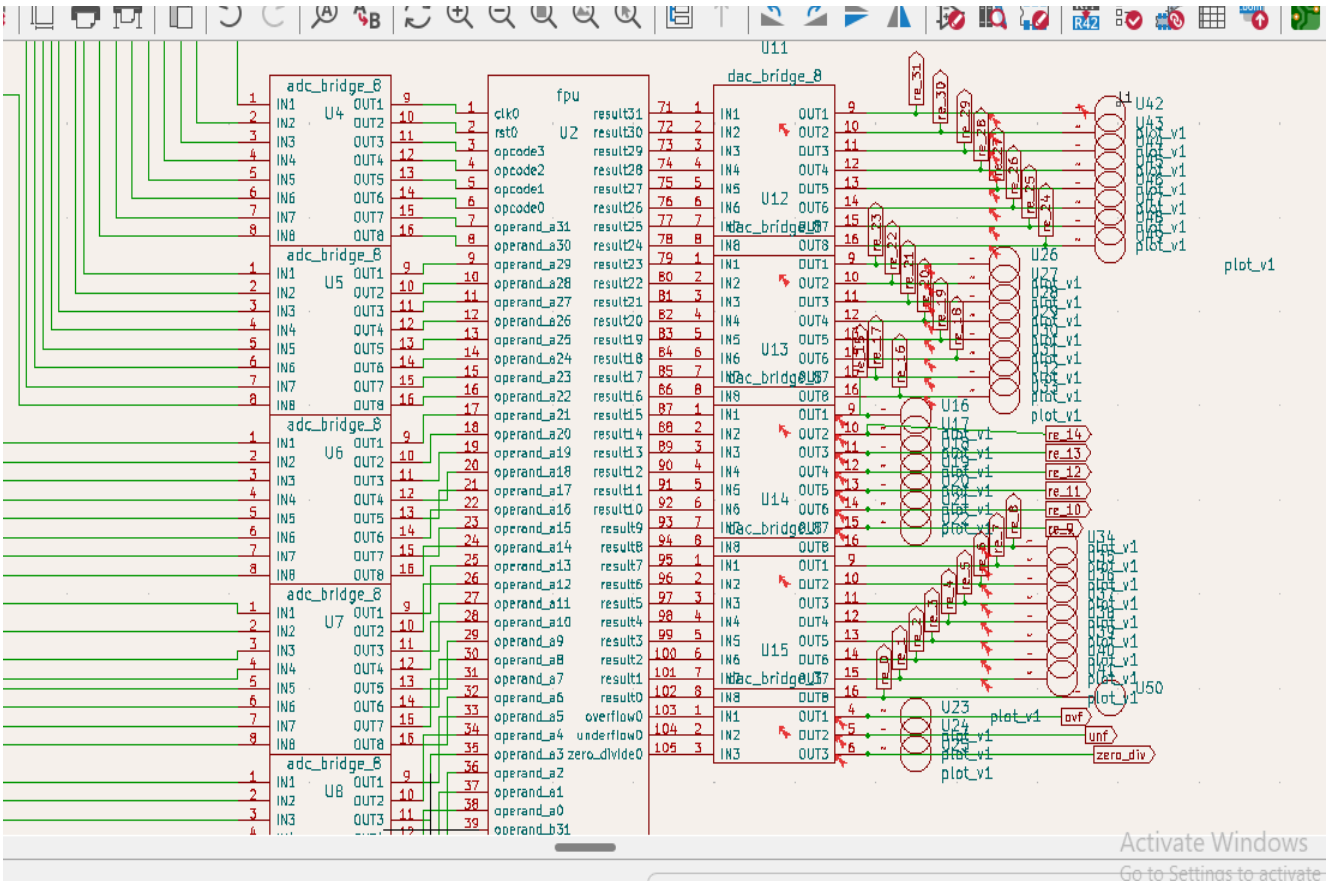


division: the output obtained is `01000000100000000000000000000000`.which is 5.0 in decimal.

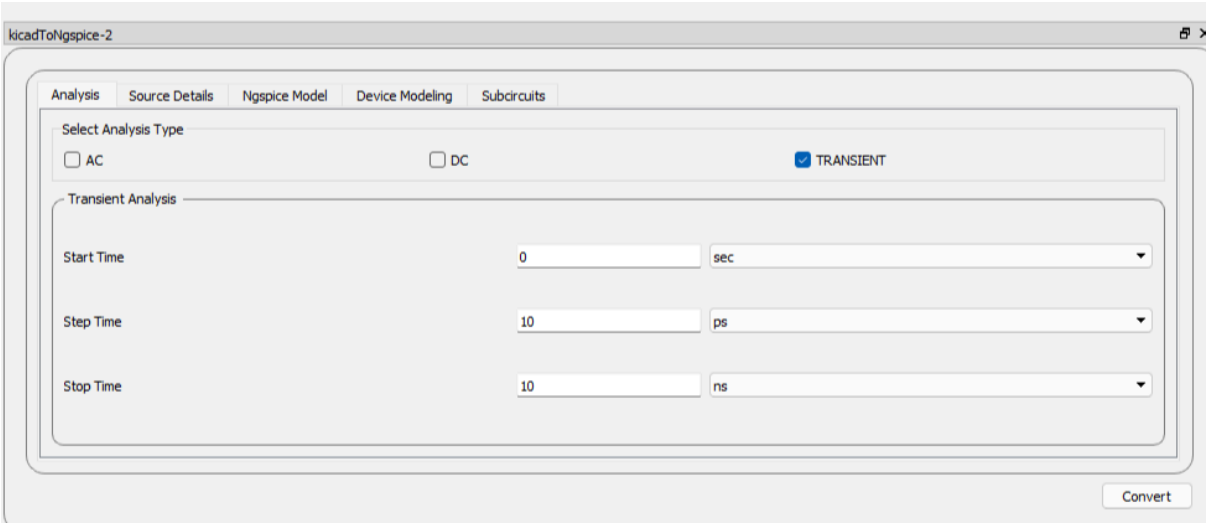


Circuit Diagram(s) :

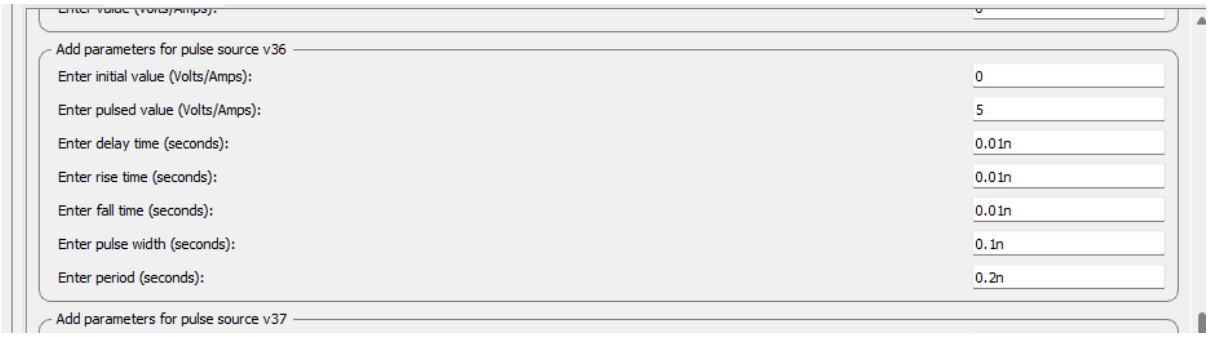
Schematic:



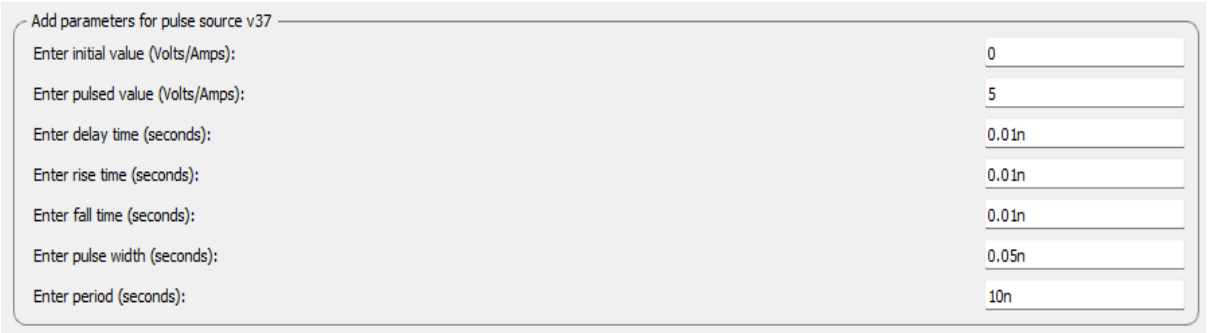
Transient Analysis:



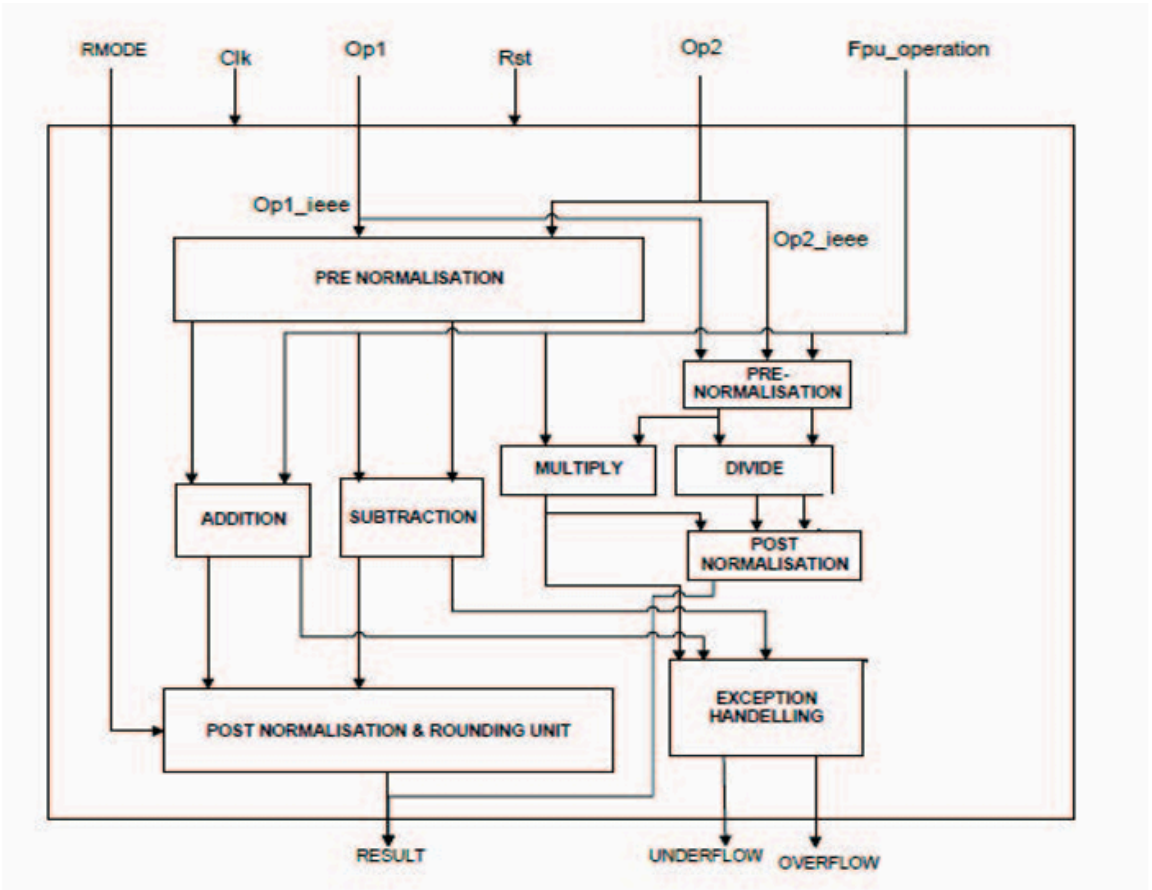
clock parameter:



Reset parameter:



Block Diagram (s) :



Expected Results (Input, Output waveforms and/or Multimeter readings) :

1. Data from NGSpice Console Logs

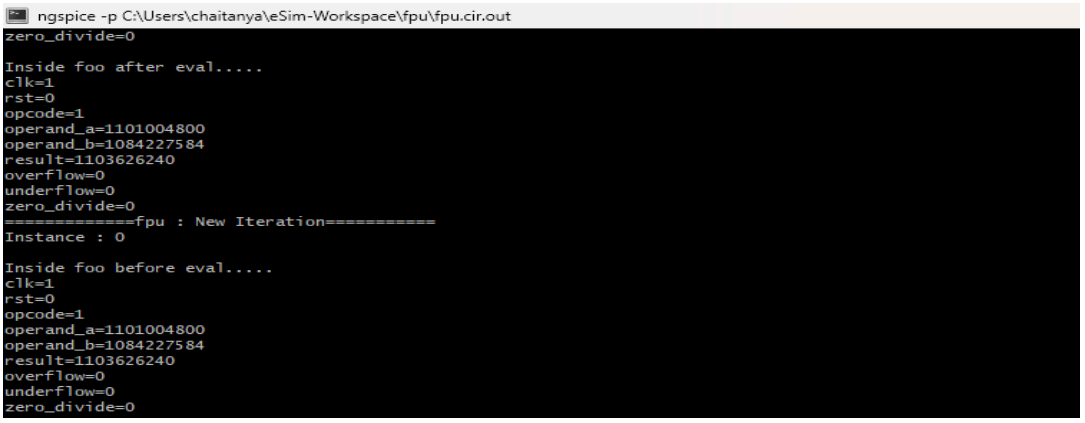
In these simulations, the inputs are provided as decimal integers representing the 32-bit floating-point data. All four operations use the same input values.

Common Inputs:

- Operand A (Input 1):
  - Decimal (Log): 1101004800
  - Binary: 01000001101000000000000000000000
  - Float Value: 20.0
- Operand B (Input 2):
  - Decimal (Log): 1084227584
  - Binary: 01000000101000000000000000000000
  - Float Value: 5.0

Addition: output obtained is

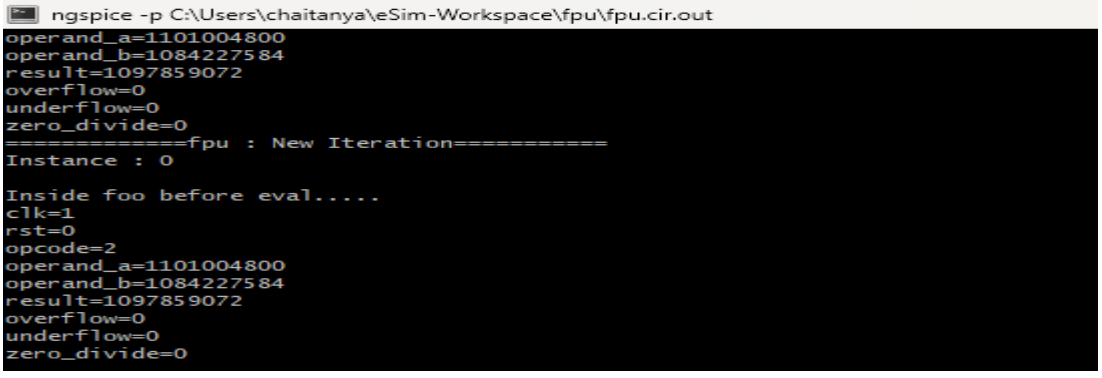
  - Decimal (Log): 1103626240
  - Binary: 01000001110010000000000000000000
  - Float Value: 25.0



subtraction:

Decimal (Log): 1097859072

  - Binary:01000001011100000000000000000000
  - Float Value: 15.0



Multiplication:

Decimal (Log): 1120403456

- **Binary:01000010110010000000000000000000**
- **Float Value: 100.0**

```
ngspice -p C:\Users\chaitanya\eSim-Workspace\fpu\fpu.cir.out
operand_a=1101004800
operand_b=1084227584
result=1120403456
overflow=0
underflow=0
zero_divide=0
============fpu : New Iteration===========
Instance : 0

Inside foo before eval.....
clk=1
rst=0
opcode=3
operand_a=1101004800
operand_b=1084227584
result=1120403456
overflow=0
underflow=0
zero_divide=0
```

Division:

Decimal (Log): 1082130432

- **Binary:01000000100000000000000000000000**
- **Float Value: 4.0**

```
ngspice -p C:\Users\chaitanya\eSim-Workspace\fpu\fpu.cir.out
operand_a=1101004800
operand_b=1084227584
result=1082130432
overflow=0
underflow=0
zero_divide=0
============fpu : New Iteration===========
Instance : 0

Inside foo before eval.....
clk=1
rst=0
opcode=4
operand_a=1101004800
operand_b=1084227584
result=1082130432
overflow=0
underflow=0
zero_divide=0
```

Research Paper/Journal/etc. : *(Providing at least one relevant research paper, journal article, or patent reference is mandatory. The reference should be directly related to the proposed circuit. Proposals submitted without a valid reference will not be accepted.)

Title : Design of Single Precision Floating Point Arithmetic logic Unit

Author :Nisha singh and R.Dhanabal

Link : Design of Single Precision Floating Point Arithmetic Logic Unit | IEEE Conference Publication | IEEE Xplore

Source/Reference(s) :Design of single precision floating point arthemetic logic unit by Nisha singh and R.Dhanbal ,VIT VELLORE(IEE)

Note: Fields marked with an asterisk (*) are mandatory and must be filled for successful submission.