OAK RIDGE National Laboratory

Exploring Application-Specific Integrated Circuit (ASIC) for EMT Simulation

Weiping Shi, Texas A&M University (wshi@ece.tamu.edu)

Zhenrui Wang, Texas A&M University (jerrywang95@tamu.edu)

Karthikeyan Balasubramaniam, Argonne National Laboratory (kbalasubramaniam@anl.gov)

This work is supported by U.S. Department of Energy (DOE) Solar Energy Technologies Office (SETO), Project #38455.

U.S. DEPARTMENT OF ENERGY

Outline

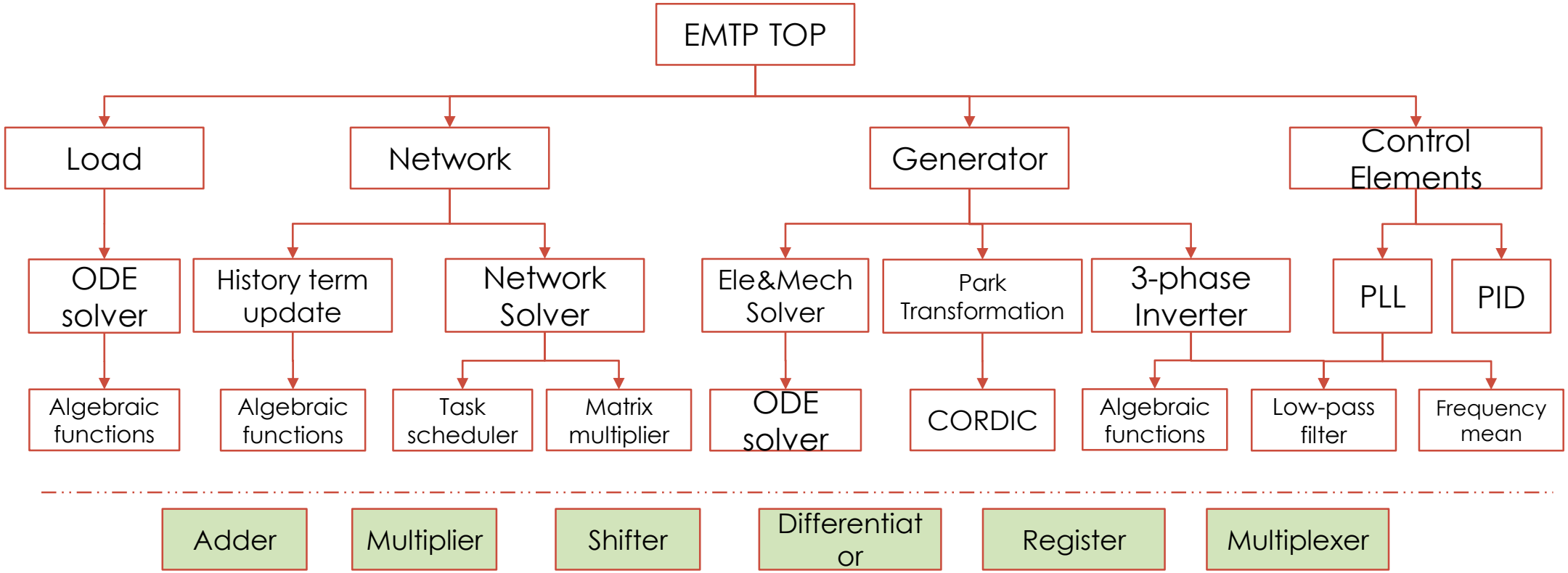
- Overview
- Hardware Accelerator Design
- Circuit Synthesis Results
- Component Design
- Conclusions

Overview

- Why?
  - Let us look at a real-world example from a different field
  - Etched --- a company founded just two years ago that designs chips for Transformer inference which are 20x faster than NVIDIA H100 GPUs.
  - "Sohu is an order of magnitude faster and cheaper than even NVIDIA's next-generation Blackwell GPUs“
  - And yes, Etched, as the name suggests is an ASIC that implements Transformers at the hardware level – “The world’s first transformer ASIC”

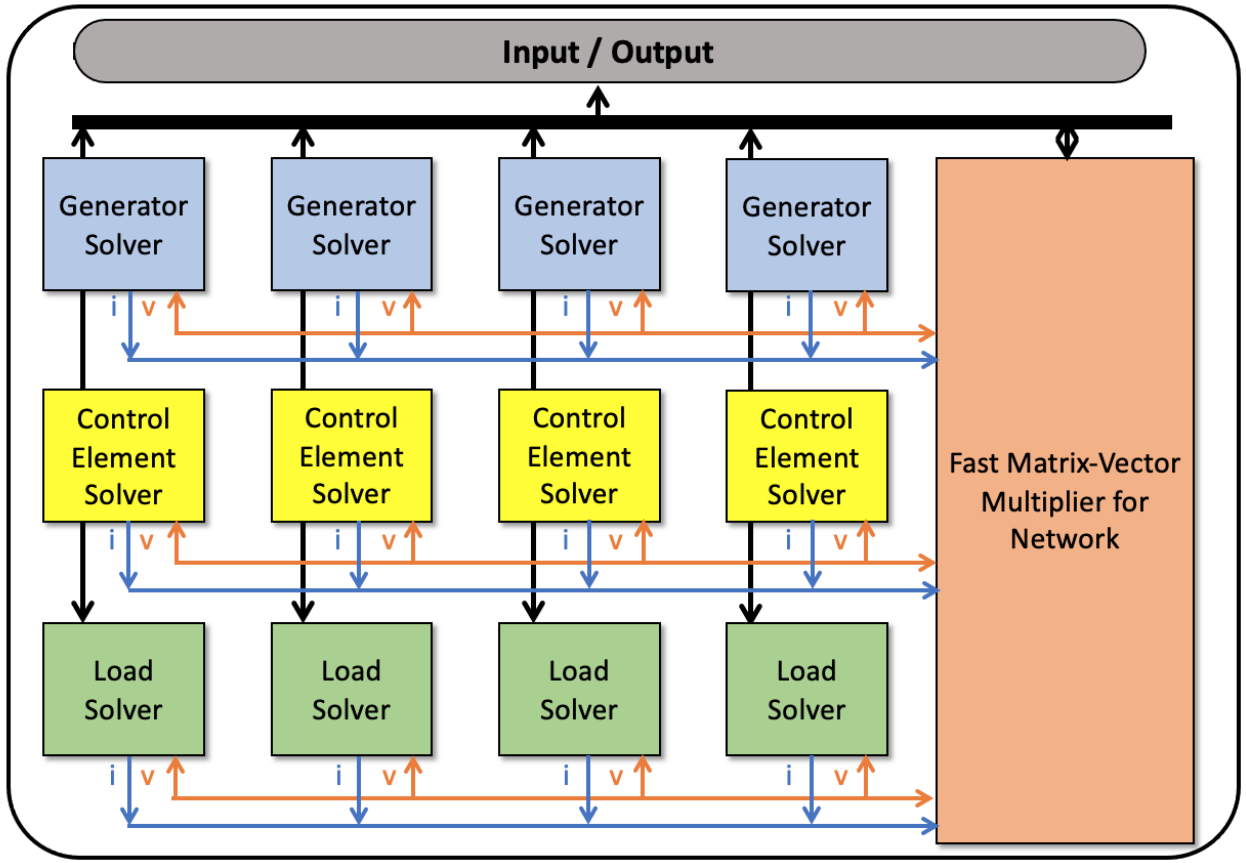
Overview

- Hardware implementation of the EMT simulation platform
  - ➢ Hierarchical structure of the components in the system



Hardware Accelerator Design

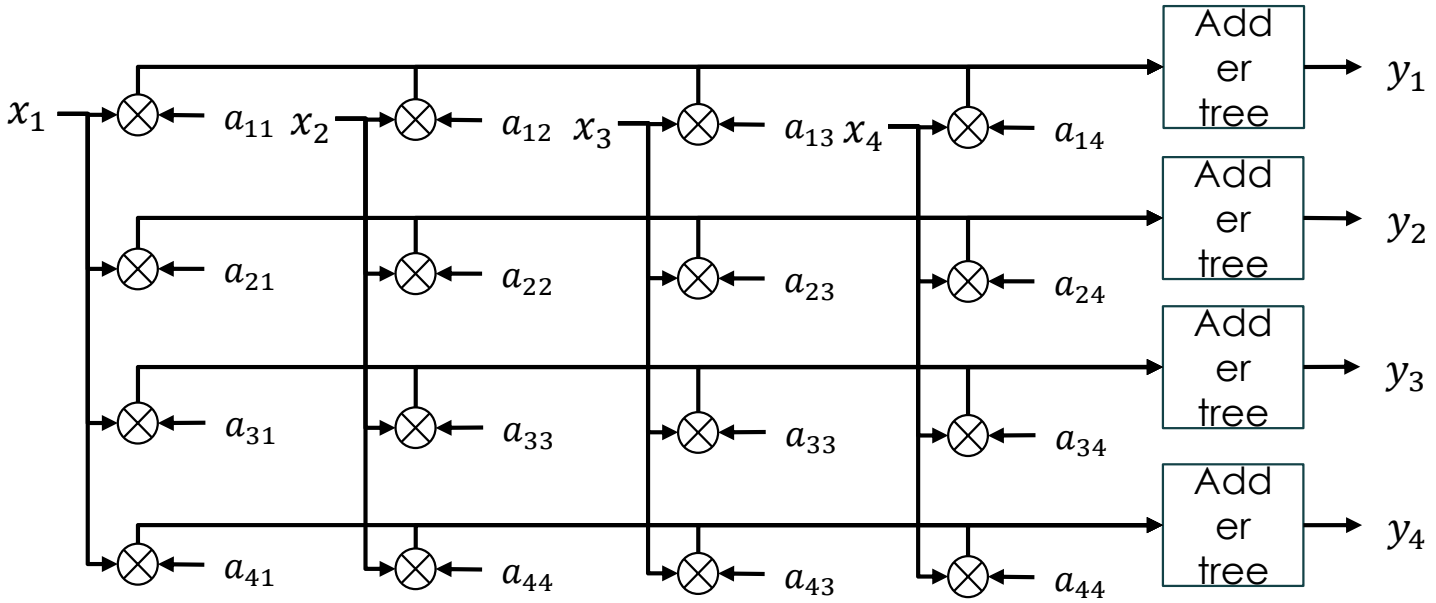
- Hardware implementation of the EMT simulation platform
  - ➢ Overall hardware architecture



- Dedicated hardware modules for network, generator, loads and control elements
- High speed buses for data transfer between network solver and other elements.
- The Matrix-vector multiplier is the key component in the architecture.

Hardware Accelerator Design

- Design of the Processing Element(PE)
  - The Processing Element(PE) is responsible for dense matrix-vector multiplication of a small tile (e.g. 4x4).
  - We utilize multiple multipliers to conduct the multiplications simultaneously.
  - Results in one row will be summed up through a set of adder tree.

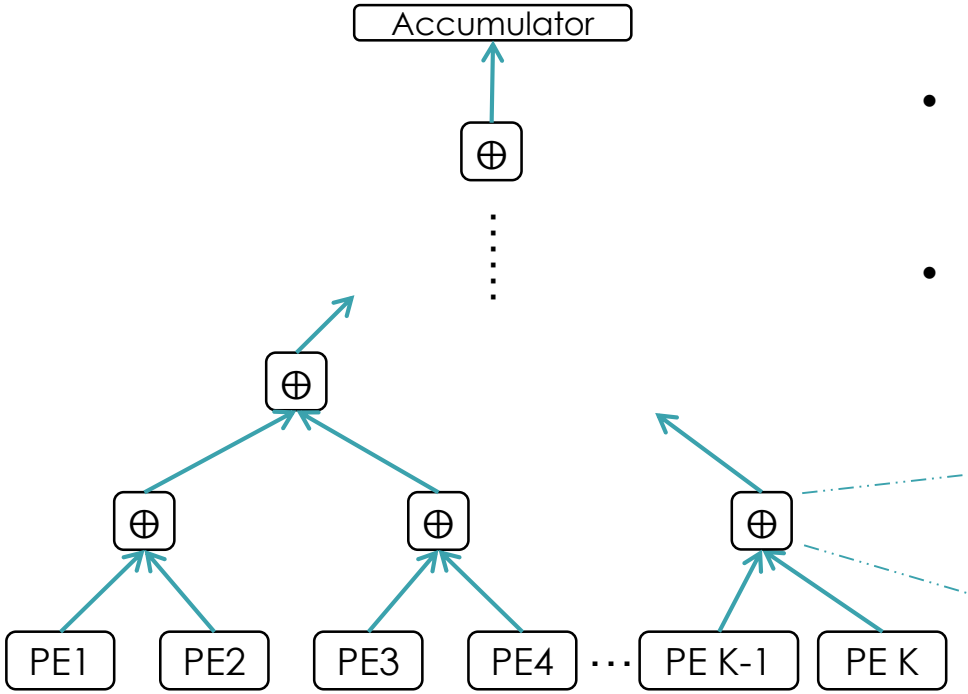


A PE architecture that conduct a 4x4 dense mat-vec (y=Ax)

- Hardware cost:
  16 multipliers
  12 adders
- Delays: 3 cycles

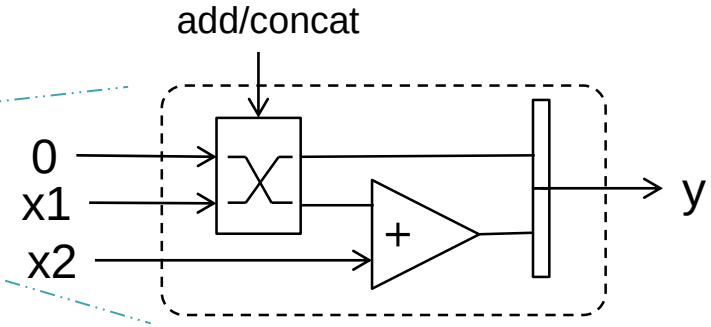
Hardware Accelerator Design

- Adaptive matrix multiplier with configurable adder tree



Configurable adder tree architecture

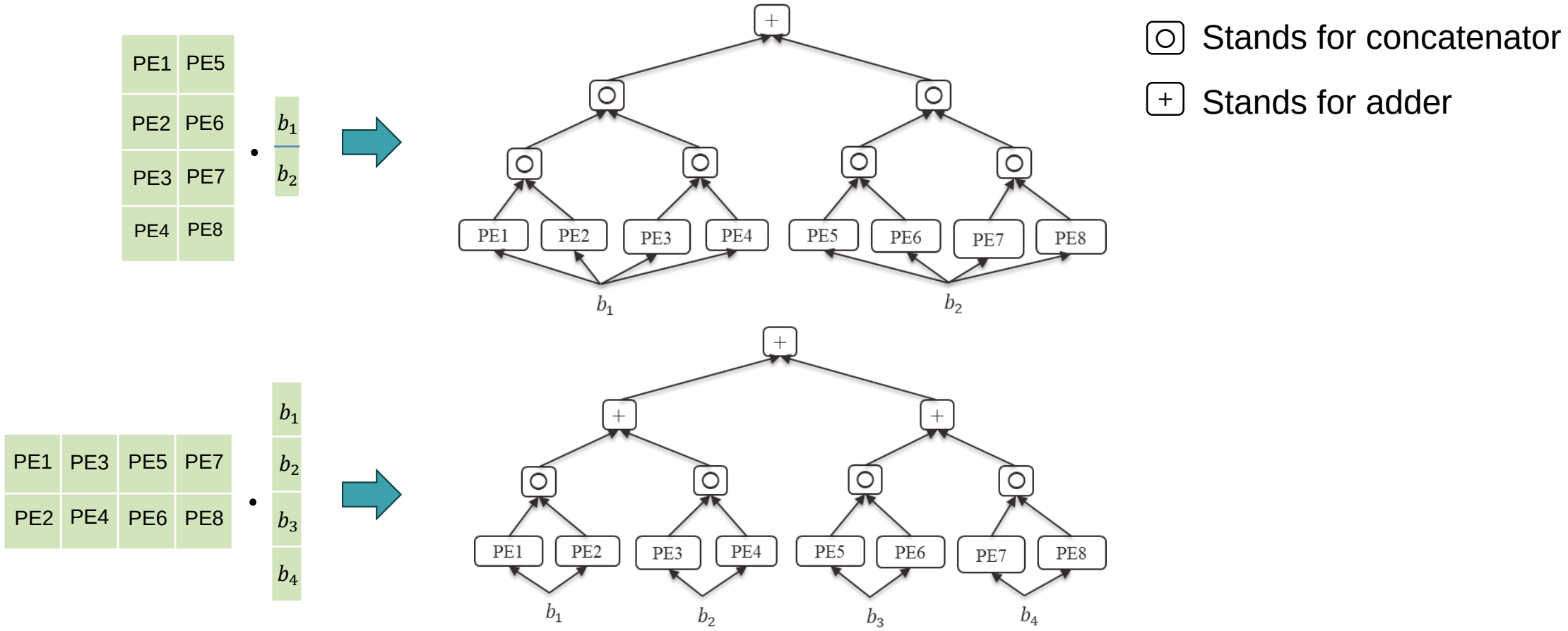
- Each PE is responsible for n*n(e.g. 4x4) dense matvec and produces a vector of length n
- ⊕ stands for a configurable unit which act as numerical adder or concatenator



Configurable adder unit

Hardware Accelerator Design

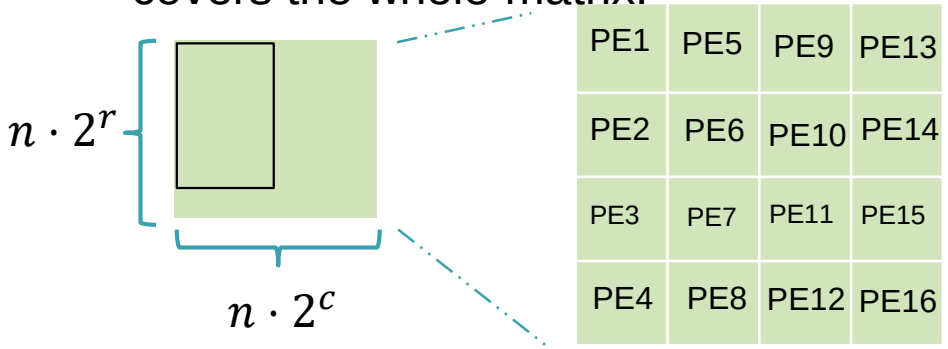
- Mapping the matrix to the architecture
  - Examples of 2 mat-vec of different shapes mapped to the same architecture



Hardware Accelerator Design

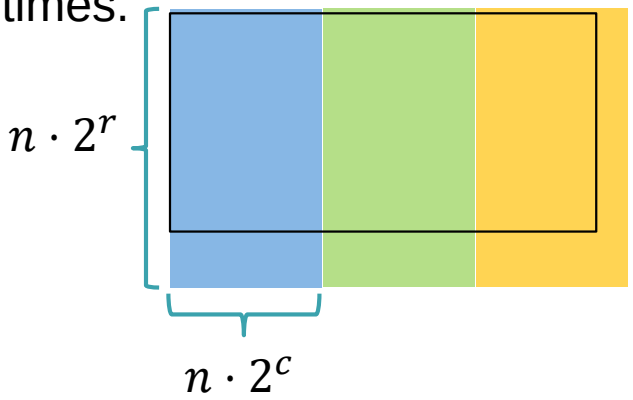
- Mapping the matrix to the architecture
  - ➢ Arrange the PEs to form an array
    - Assume the total number of PEs is 2^k(k is the number of levels of the tree).
    - The possible shape of PE array could be $(n \cdot 2^r, n \cdot 2^c)$, where r+c=k, n is the PE tile size.
  - ➢ Shaping strategy of the PE array
    - **Case 1** (sufficient amount of PEs)

      Shape the PE array so that it covers the whole matrix.



    - **Case 2** (insufficient amount of PEs)

      Shape the PE array so that it covers part of the columns of the matrix. The whole matrix is covered in multiple times.



Circuit Synthesis Results

- Circuit synthesis results
  - Synthesis tool: Synopsys Design Compiler
  - **Target library: Nangate open cell library (45nm)**
  - Circuit parameters: Number of PEs: 32, PE tile size: 8, Word width: 8 (enough to contain 69-bus system)

➢ Power report

Power Group	Internal Power	Switching Power	Leakage Power	Total Power	(%)
io_pad	0.0000	0.0000	0.0000	0.0000	(0.00%)
memory	0.0000	0.0000	0.0000	0.0000	(0.00%)
black_box	0.0000	0.0000	0.0000	0.0000	(0.00%)
clock_network	2.2662e+04	0.0000	0.0000	2.2662e+04	(30.58%)
register	5.8182e+03	1.6141e+03	3.3552e+06	1.0788e+04	(14.56%)
sequential	0.0000	0.0000	0.0000	0.0000	(0.00%)
combinational	1.7179e+04	1.1350e+04	1.2117e+07	4.0648e+04	(54.86%)
Total	4.5660e+04 uW	1.2964e+04 uW	1.5472e+07 nW	7.4098e+04 uW	

Power consumption < 0.08 w

➢ Area report

```
Combinational area:            542148.701792
Buf/Inv area:                   81716.530028
Noncombinational area:         194482.168945
Macro/Black Box area:               0.000000
Net Interconnect area:      undefined  (Wire load has zero net area)

Total cell area:               736630.870737
```

Total area < 0.8 mm^2

Circuit Synthesis Results

- Circuit performance evaluation
  - Based on synthesis result on timing, we can estimate the time needed for the accelerator to solve the 69-bus system
  - Circuit parameters: Number of PEs: 32, PE tile size: 8, Word width: 8 (sufficient to process the 69-bus system)

➢ Timing report

```
clock clk (rise edge)                      10.00      10.00
clock network delay (ideal)                 0.00      10.00
tree_genblk1_8__adder_cater_out_reg_0_/CK   0.00      10.00 r
library setup time                         -0.04       9.96
data required time                                     9.96
-----------------------------------------  ----------------
data required time                                     9.96
data arrival time                                     -1.28
-----------------------------------------  ----------------
slack (MET)                                            8.69
```

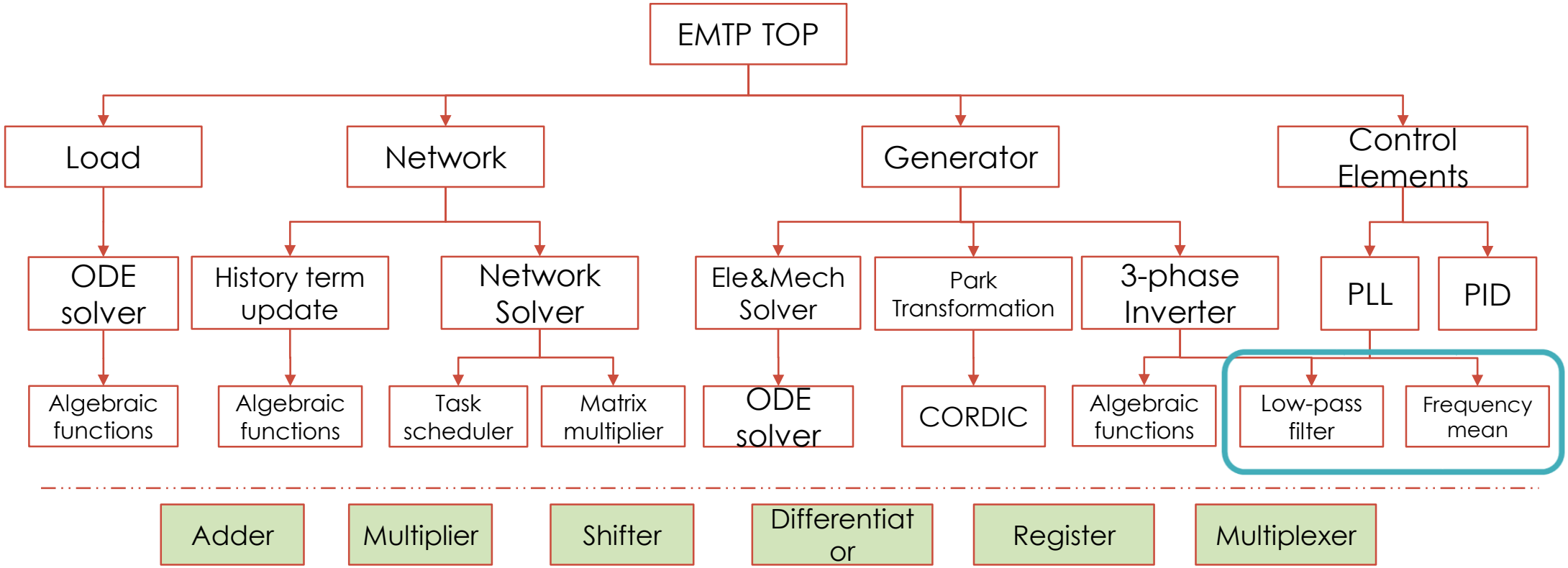
Running frequency > 500Mhz

➢ Runtime estimation

- Delay for each mat-vec: $\log 8 + \log 32 + 4 = 12$ cycles.
- To solve the linear system we need 4 mat-vec, thus total delay: 4*12=48 cycles.
- Assume running frequency 500 MHz, total runtime would be 48*2 ns = 96 ns.
- Compared to previous software approach (~10 us), hardware accelerator gains around 104 times speedup for the 69-bus system.

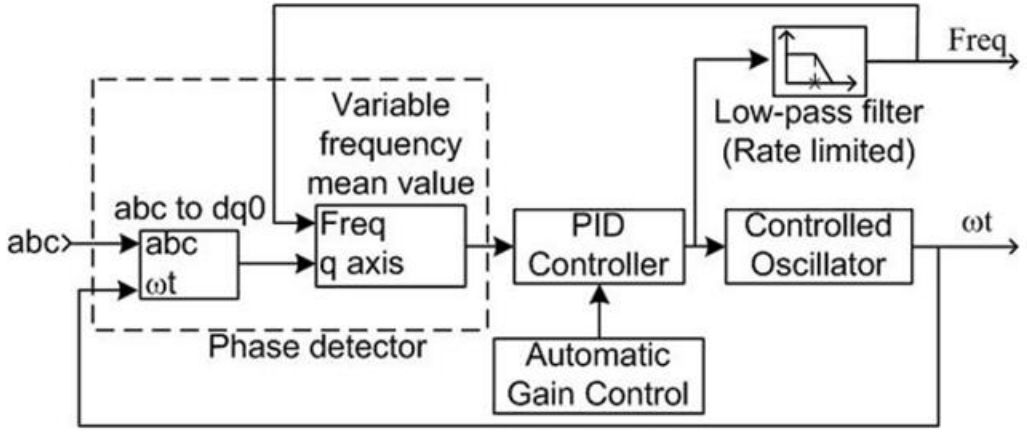
Component Design

- Hardware implementation of the EMT simulation platform
  - ➢ Hierarchical structure of the components in the system



PLL Design -- Variable Frequency Mean Function

- Hardware implementation of the variable frequency mean function

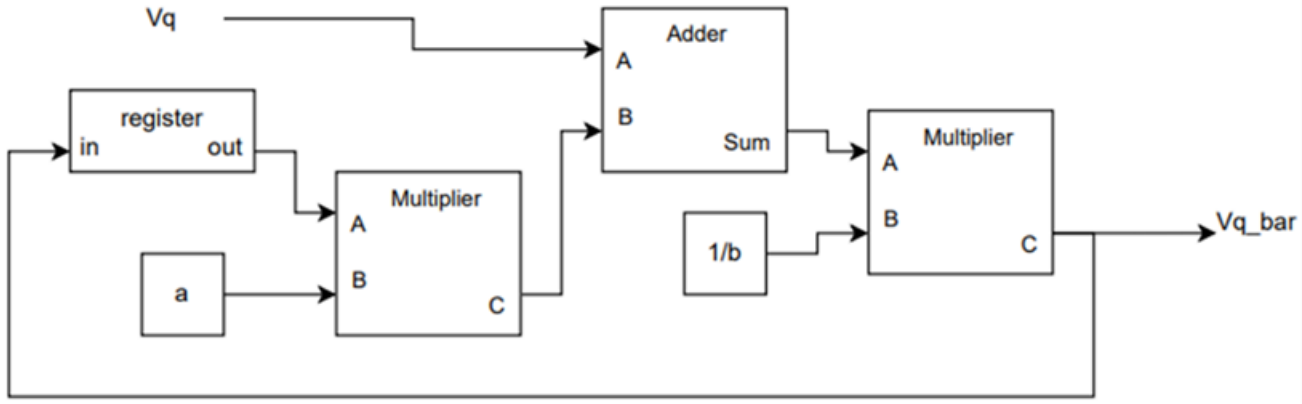


- The variable frequency mean function calculate the mean value of the input signal over a single cycle.

$$\bar{V}(t) = \frac{1}{T}\int_{t-T}^{t} V(t) \cdot dt$$

PLL Design -- Variable Frequency Mean Function

- Hardware implementation of the variable frequency mean function
  - The running mean form can be implemented very efficiently with minimum hardware requirement.



- 2 multiplier
- 1 adder
- 1 register

PLL Design -- Low-Pass Filter

- Hardware implementation of the low-pass filter
  - The low-pass filter can remove high-frequency noises thus makes the output frequency smoother and more robust.
  - The general form of the low-pass filter can be expressed as:

$$f'' + 2\xi\omega_n \cdot f' + \omega_n \cdot f = f(t)$$

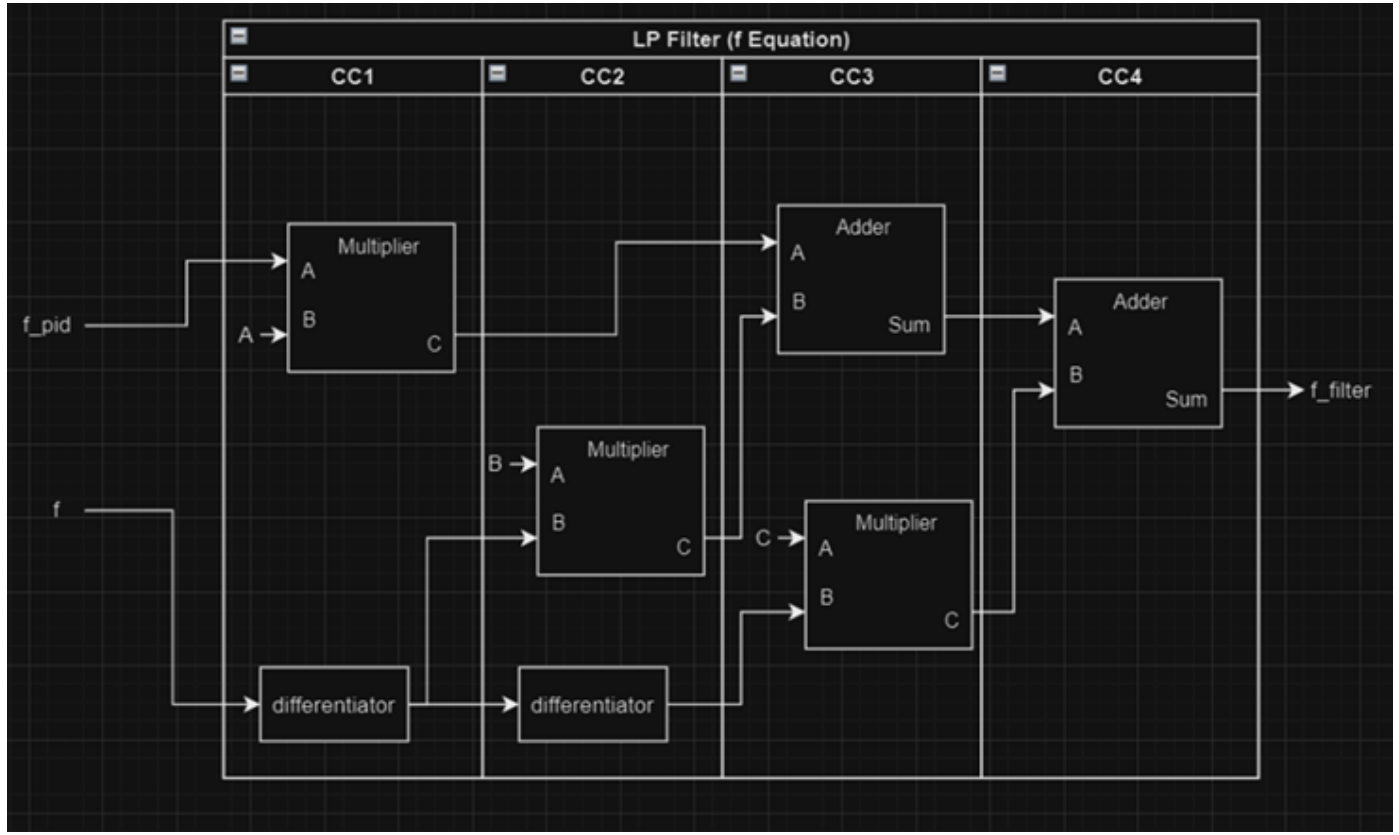
  - To facilitate hardware implementation, we rewrite the equation as:

$$f = \frac{f(t)}{2\pi} - \frac{f''}{\omega_n^2} - \frac{2\xi f'}{\omega_n} = A \cdot f(t) - B \cdot f' - C \cdot f''$$

Now we only need to calculate the derivative and second-order derivative of f, which can be easily obtained by the difference between current value and history values.

PLL Design -- Low-Pass Filter

- Hardware implementation of the low-pass filter
  - ➢ The design that maximize hardware reuse



Hardware count
- Multiplier: 1
- Adder: 1
- Differentiator: 1
  - Adder: 0 (will use available adders)
  - Shifter: 1
  - Register: 2

Timing
- 4 Cycles
- 1 cycle has delay of 1 multiplier

Another option of the same circuit performs CC3 and CC4 in one cycle.
- Increases hardware usage
- Decreases #CCs

Conclusions

- ASIC for EMT research is complementary to existing EMT research.
- The research in this space is currently in early stage but has tremendous potential for accelerating EMT studies, drawing inspiration from success stories in other fields.

Questions?