भारतीय सूचना प्रौद्योगिकी संस्थान, नागपुर
Indian Institute of Information Technology, Nagpur
Institute of National Importance by the Act of Parliament
The Indian Institute of Information Technology (PPP) Act, 2017

Department of Electronics and Communication Engineering

Complementary Metal-Oxide Semiconductor (CMOS)

Project Title: Implementation of 2 Bit Vedic Multiplier using CMOS Technology

Contents

- What is Vedic Multiplier
- Block Diagram of 2 Bit Vedic Multiplier
- Optimized Circuit Diagram
- MicroWind Layouts
- Netlist
- Outputs
- Advantages
- Applications

Name: Aneket Burman
Roll No: BT22ECE133
Subject: CMOS
Submitted To: Dr. Paritosh Peshwe

What is a Vedic Multiplier?

The Vedic Multiplier is a high-speed, low-power multiplier architecture based on ancient Indian Vedic mathematics, particularly the "Urdhva Tiryakbhyam" (vertical and crosswise) sutra. Unlike traditional multiplication techniques that rely on sequential addition and shifting, the Vedic method allows parallel generation of partial products, resulting in reduced computation time and improved efficiency, making it ideal for applications in digital signal processing, image processing, and low-power embedded systems.

Key Components and Design of a Vedic Multiplier

The Vedic multiplier architecture is built on hierarchical and modular principles, enabling the design of fast and scalable multipliers for both small and large bit-width operations. Its main components include:

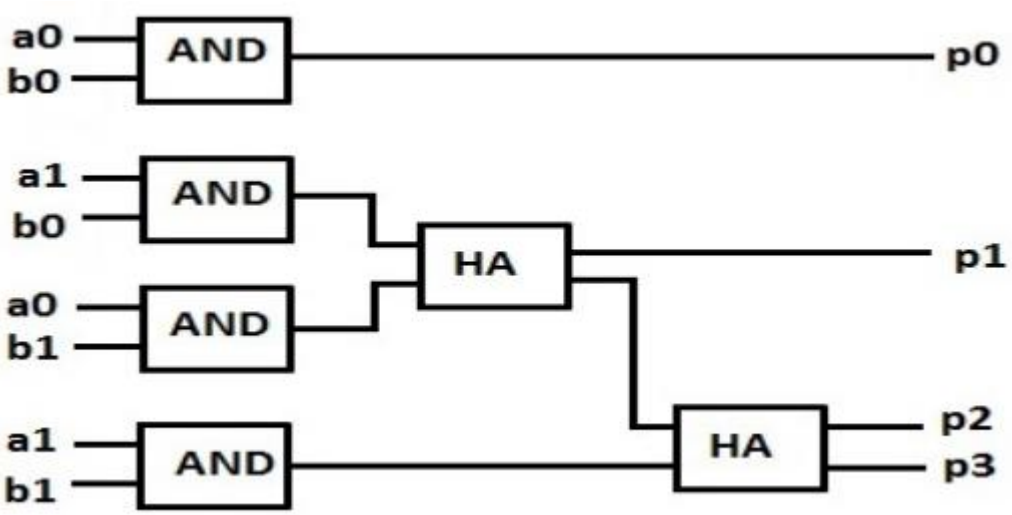
1. **Urdhva Tiryakbhyam Sutra**:
   This ancient technique is the foundational logic for the Vedic multiplier. It performs multiplication in a manner similar to convolution, where the partial products are calculated in parallel using vertical and crosswise operations. This allows for a significant reduction in propagation delay compared to conventional algorithms like Booth or array multipliers.
2.
2. **Partial Product Generator (PPG)**:
   The PPG block computes all possible partial products between bits of the multiplicand and multiplier using simple AND gates. For instance, in a 2x2 multiplication, four partial products are generated simultaneously: A0B0, A0B1, A1B0, and A1B1.

3. **Adder Units**:
   The adder blocks are used to accumulate the partial products generated by the PPG. These can be Ripple Carry Adders, Carry Save Adders, or other fast adder architectures depending on the design goals (speed, area, power). In larger bit-width multipliers like 4x4 or 8x8, multiple adder stages are organized hierarchically.

4. **Recursive or Modular Design**:
   Higher-order Vedic multipliers (e.g., 4x4, 8x8, 16x16) are built recursively by combining several smaller Vedic multiplier blocks. For example, a 4x4 multiplier can be constructed using four 2x2 multipliers, adders, and shifters. This modularity makes Vedic multipliers highly scalable and reusable.

5. **Optimized Delay Path**:
   The key advantage of the Vedic method is its minimal critical path delay. As partial products are generated and added in parallel, the overall multiplication time is reduced significantly compared to serial techniques.

Block Diagram of 2 Bit Vedic Multiplier



Example:

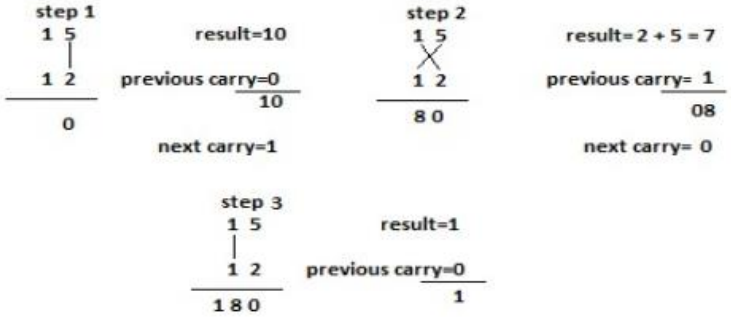
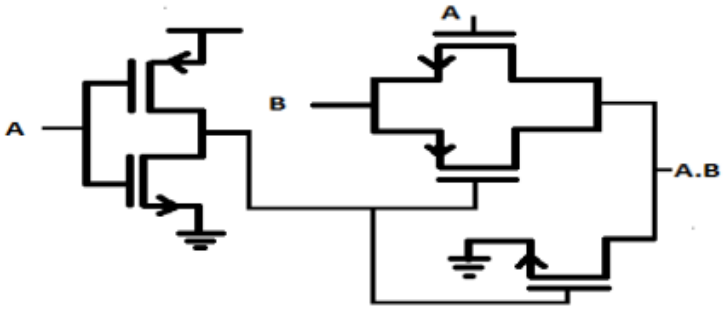


Fig. 1 UT based multiplication for 2 digit decimal numbers

Optimized AND Gate (5T Design)

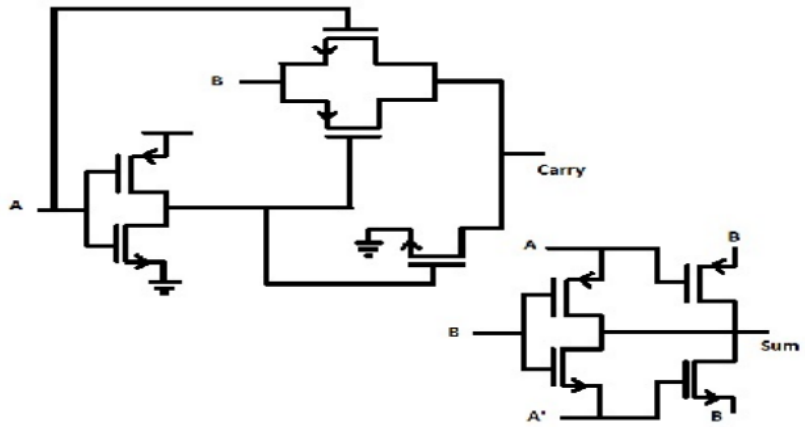


The project employs a 5-transistor (5T) AND gate for partial product generation, leveraging pass transistor logic and transmission gate-based logic. This design is multiplexer-inspired, where the output behaves as:

- B when A = 1, and
- 0 (grounded) when A = 0,
  thus realizing the logical A AND B function.

Compared to the conventional 6-transistor CMOS implementation, this design saves one transistor per gate, leading to significant area optimization, especially in higher-bit multipliers. For example, an 8-bit multiplier requires 64 AND gates, so the 5T design reduces transistor count by 64 transistors per multiplier unit. Since partial products are generated in parallel, this optimization does not significantly impact delay characteristics.

Optimized Half Adder (9T Design)

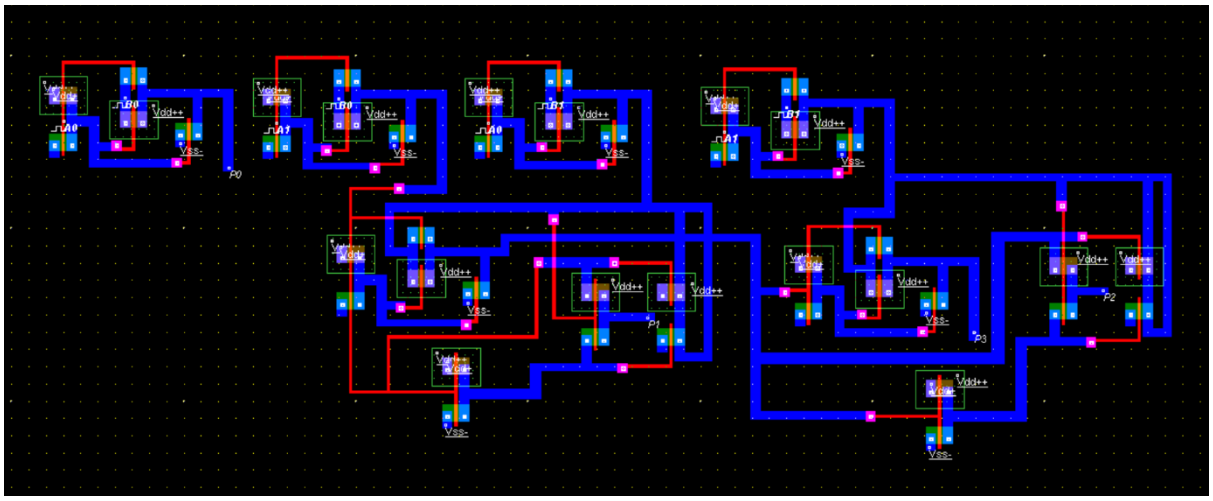


The 9-transistor (9T) half adder used in this project emphasizes high speed, low power, and full swing operation. It consists of:

- A 5T AND gate for carry generation (as discussed above), and
- A 4T XOR gate for sum generation.

This compact design offers a reduced transistor count, leading to lower power dissipation and smaller chip area. The full swing logic levels ensure reliable operation even in cascaded configurations, eliminating the need for extra buffers or swing restoration circuits. Additionally, the reduced delay in this topology contributes to enhanced performance in arithmetic units.

MicroWind Layout



Netlist-

*** POWER SUPPLY SOURCES ***
VDD 1 0 DC 2.00
V2 2 0 DC 2.50
V3 3 0 DC 2.50
V4 4 0 DC 2.50
V5 5 0 DC 2.50
V6 6 0 DC 2.50
V7 7 0 DC 2.50
V8 8 0 DC 2.50
V9 9 0 DC 2.50
V10 10 0 DC 2.50
V11 11 0 DC 2.50
V12 12 0 DC 2.50
V13 13 0 DC 2.50
V14 14 0 DC 2.50
V15 15 0 DC 2.50
V16 16 0 DC 2.50
V17 17 0 DC 2.50
V18 18 0 DC 2.50
V19 19 0 DC 2.50

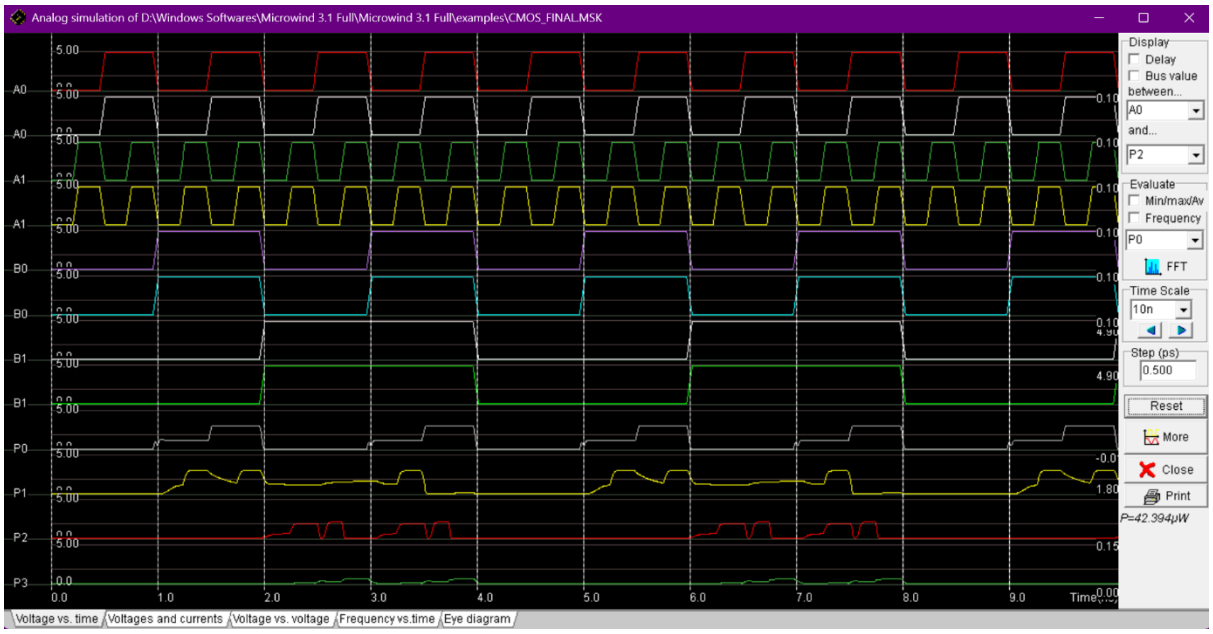
*** INPUT PULSE SOURCES ***
V_B1_1 29 0 PULSE(0 5 1.95n 0.05n 0.05n 1.95n 4n)
V_B1_2 35 0 PULSE(0 5 1.95n 0.05n 0.05n 1.95n 4n)
V_B0_1 42 0 PULSE(0 5 0.95n 0.05n 0.05n 0.95n 2n)
V_B0_2 46 0 PULSE(0 5 0.95n 0.05n 0.05n 0.95n 2n)

```
V_A0_1 65 0 PULSE(0 5 0.45n 0.05n 0.05n 0.45n 1n)
V_A1_1 66 0 PULSE(0 5 0.20n 0.05n 0.05n 0.20n 0.5n)
V_A0_2 67 0 PULSE(0 5 0.45n 0.05n 0.05n 0.45n 1n)
V_A1_2 68 0 PULSE(0 5 0.20n 0.05n 0.05n 0.20n 0.5n)

*** NMOS TRANSISTORS ***
MN1 20 30 0 0 N1 L=0.2u W=1u
MN2 26 23 20 0 N1 L=0.2u W=1u
MN3 26 27 0 0 N1 L=0.2u W=1u
MN4 24 23 0 0 N1 L=0.2u W=1u
MN5 22 20 24 0 N1 L=0.2u W=1u
MN6 49 24 20 0 N1 L=0.2u W=1u
MN7 53 23 27 0 N1 L=0.2u W=1u
MN8 20 68 29 0 N1 L=0.2u W=1u
MN9 30 68 54 0 N1 L=0.2u W=1u
MN10 23 38 55 0 N1 L=0.2u W=1u
MN11 23 36 0 0 N1 L=0.2u W=1u
MN12 33 23 38 0 N1 L=0.2u W=1u
MN13 23 67 35 0 N1 L=0.2u W=1u
MN14 36 67 57 0 N1 L=0.2u W=1u
MN15 23 40 0 0 N1 L=0.2u W=1u
MN16 38 34 0 0 N1 L=0.2u W=1u
MN17 34 43 0 0 N1 L=0.2u W=1u
MN18 40 34 61 0 N1 L=0.2u W=1u
MN19 34 66 42 0 N1 L=0.2u W=1u
MN20 43 66 62 0 N1 L=0.2u W=1u
MN21 45 47 0 0 N1 L=0.2u W=1u
MN22 45 65 46 0 N1 L=0.2u W=1u
MN23 47 65 64 0 N1 L=0.2u W=1u

*** PMOS TRANSISTORS ***
MP1 20 23 21 14 P1 L=0.2u W=1u
MP2 22 20 23 13 P1 L=0.2u W=1u
MP3 24 23 1 19 P1 L=0.2u W=1u
MP4 26 27 20 16 P1 L=0.2u W=1u
MP5 27 23 1 15 P1 L=0.2u W=1u
MP6 20 30 29 18 P1 L=0.2u W=1u
MP7 30 68 1 17 P1 L=0.2u W=1u
MP8 23 34 32 10 P1 L=0.2u W=1u
MP9 33 23 34 11 P1 L=0.2u W=1u
MP10 23 36 35 8 P1 L=0.2u W=1u
MP11 36 67 1 9 P1 L=0.2u W=1u
MP12 38 34 1 12 P1 L=0.2u W=1u
MP13 40 34 1 7 P1 L=0.2u W=1u
```

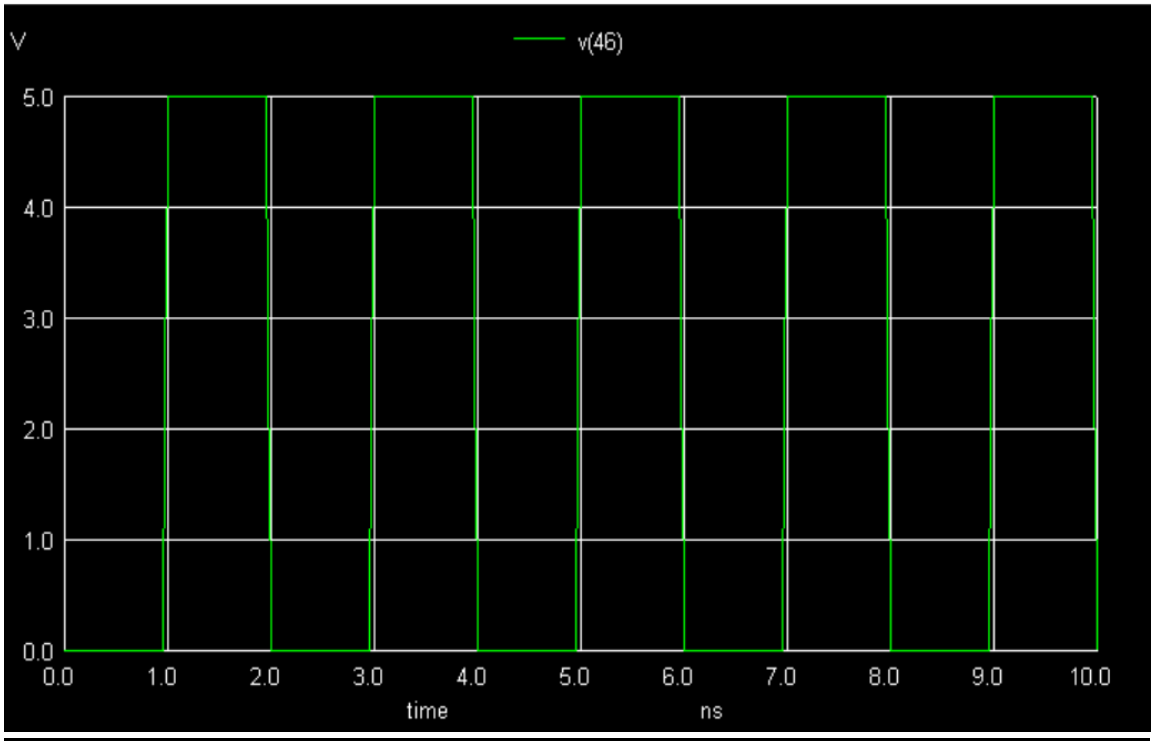
```
MP14 34 43 42 5 P1 L=0.2u W=1u
MP15 43 66 1 4 P1 L=0.2u W=1u
MP16 45 47 46 2 P1 L=0.2u W=1u
MP17 47 65 1 3 P1 L=0.2u W=1u

*** CAPACITORS ***
* Example: C2 2 0 1.540fF
C2 2 0 1.54f
C3 3 0 1.54f
C4 4 0 1.54f
...
C68 68 0 0.184f

*** NMOS MODEL***
.model N1 nmos level=14 VTH0=0.55

*** PMOS MODEL ***
.model P1 pmos level=14 VTH0=-0.55

*** ANALYSIS ***
.tran 0.01n 10n
.control
run
plot v(46)
plot v(26)
plot v(66)
plot v(20)
.endc
.end
```

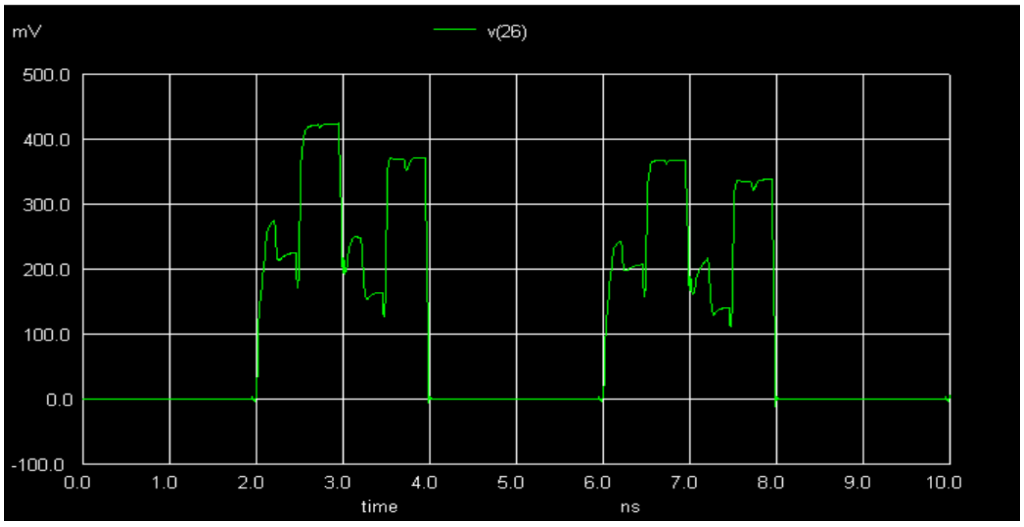
Outputs:

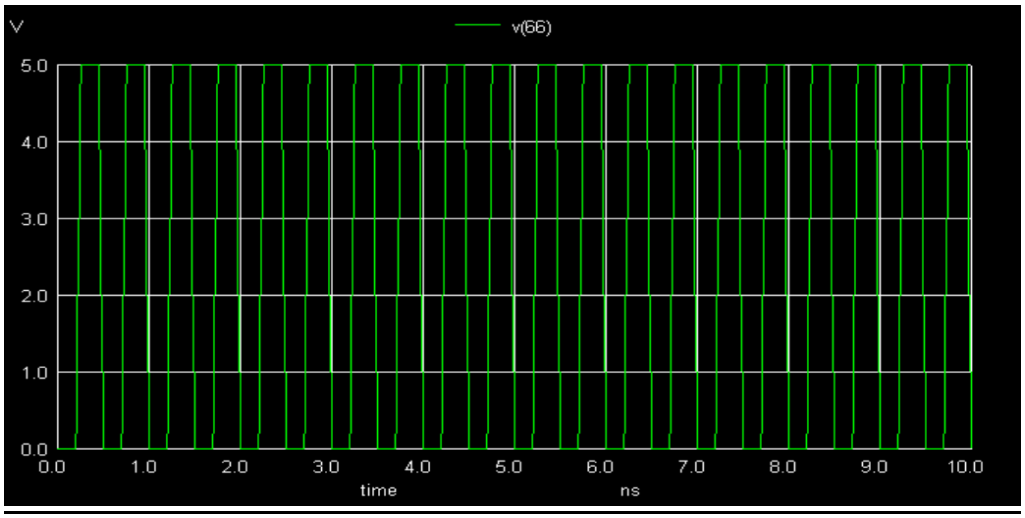
Microwind:

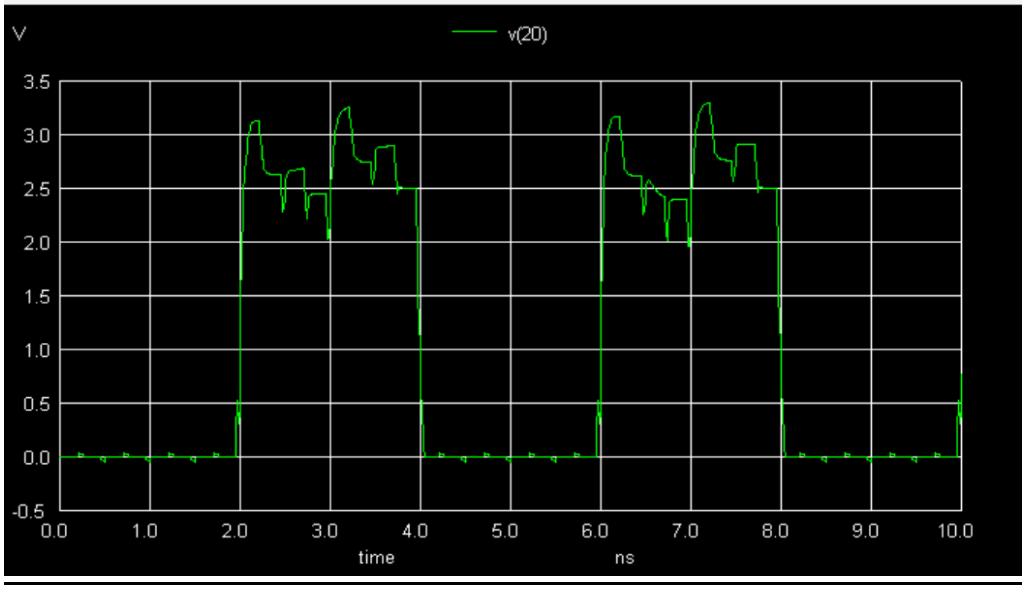


NGSpice:




mV
v(26)
500.0
400.0
300.0
200.0
100.0
0.0
-100.0
0.0 1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0
time
ns


V
v(66)
5.0
4.0
3.0
2.0
1.0
0.0
0.0 1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0
time
ns


V
v(20)
3.5
3.0
2.5
2.0
1.5
1.0
0.5
0.0
-0.5
0.0 1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0
time
ns

Advantages:

Here are some key **advantages of a 2-bit Vedic multiplier**:

1. Simplicity and Fast Computation

- The 2-bit Vedic multiplier uses **Urdhva Tiryakbhyam Sutra** from Vedic Mathematics.
- It performs **parallel generation of partial products**, which reduces computation time compared to traditional methods.

2. Low Hardware Complexity

- Due to the small bit-width, the number of logic gates required is minimal.
- Ideal for **low-power, low-area** applications like embedded systems or portable devices.

3. High Speed

- The Vedic algorithm's **regular structure** allows for faster execution with less propagation delay, especially beneficial for FPGA/ASIC implementations.

4. Scalability

- The 2-bit multiplier can be used as a **building block** for creating higher-bit multipliers (like 4-bit, 8-bit).
- Promotes a **modular design approach** for larger systems.

5. Easy to Design and Implement

- Straightforward implementation using logic gates.
- Suitable for **educational purposes and proof-of-concept** designs.

Application:

Here are some key **applications of a 2-bit Vedic multiplier**:

1. Digital Signal Processing (DSP)

- Used in simple DSP operations like filtering and convolution where low-bit multipliers are sufficient.
- Ideal for low-latency, power-sensitive DSP tasks in embedded systems.

2. Arithmetic Logic Units (ALUs)

- Serves as a basic multiplication unit inside ALUs of small processors.
- Used in applications where multiplication of small operands is required, like microcontrollers.

3. Low-Power Devices

- Perfect for **Internet of Things (IoT)** devices, smart wearables, and battery-operated gadgets that need lightweight processing with minimal power consumption.

4. Educational and Research Tools

- Great for teaching basic concepts of digital design, Vedic mathematics, and multiplication logic.
- Frequently used in labs and projects to demonstrate **efficient hardware design techniques**.

5. Image and Video Processing (Low-Resolution)

- In lightweight image processing tasks (e.g., grayscale image manipulation), 2-bit multipliers can help in pixel-based arithmetic operations.

6. FPGA/ASIC Prototyping

- Used as a fundamental module in larger designs during **prototyping or simulation** of digital circuits.
- Ideal for testing and validating Vedic math-based multiplication architectures.