RISC V Architecture

Prof. H R Vanamala
Department of Electronics and Communication Engg.

PES UNIVERSITY
CELEBRATING 50 YEARS

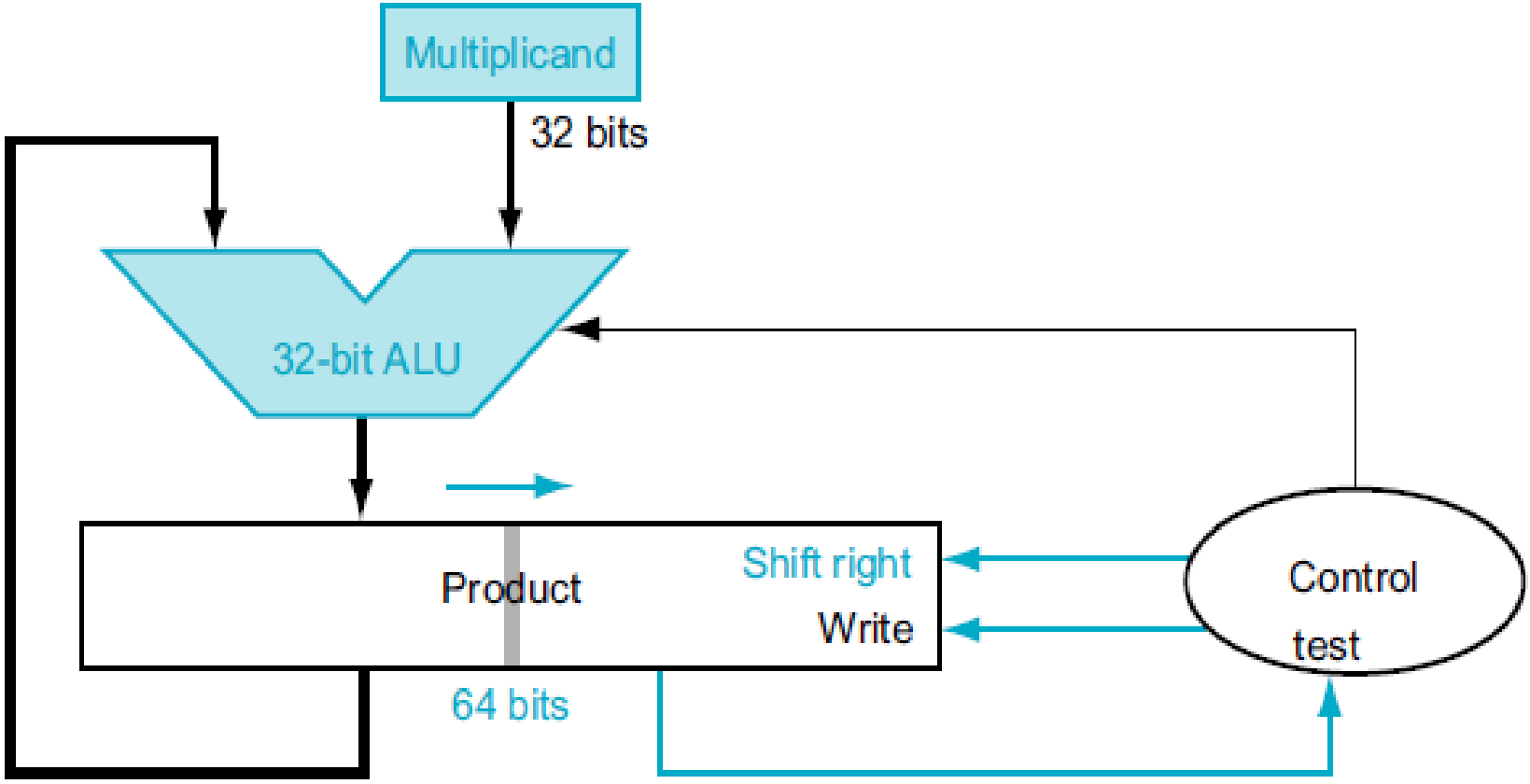
RISC V ARCHITECTURE

UNIT 4: Arithmetic for Computers

Prof. H R Vanamala

Department of Electronics and Communication Engineering

Refined version of the multiplication hardware



The changes are highlighted in color in the above diagram.

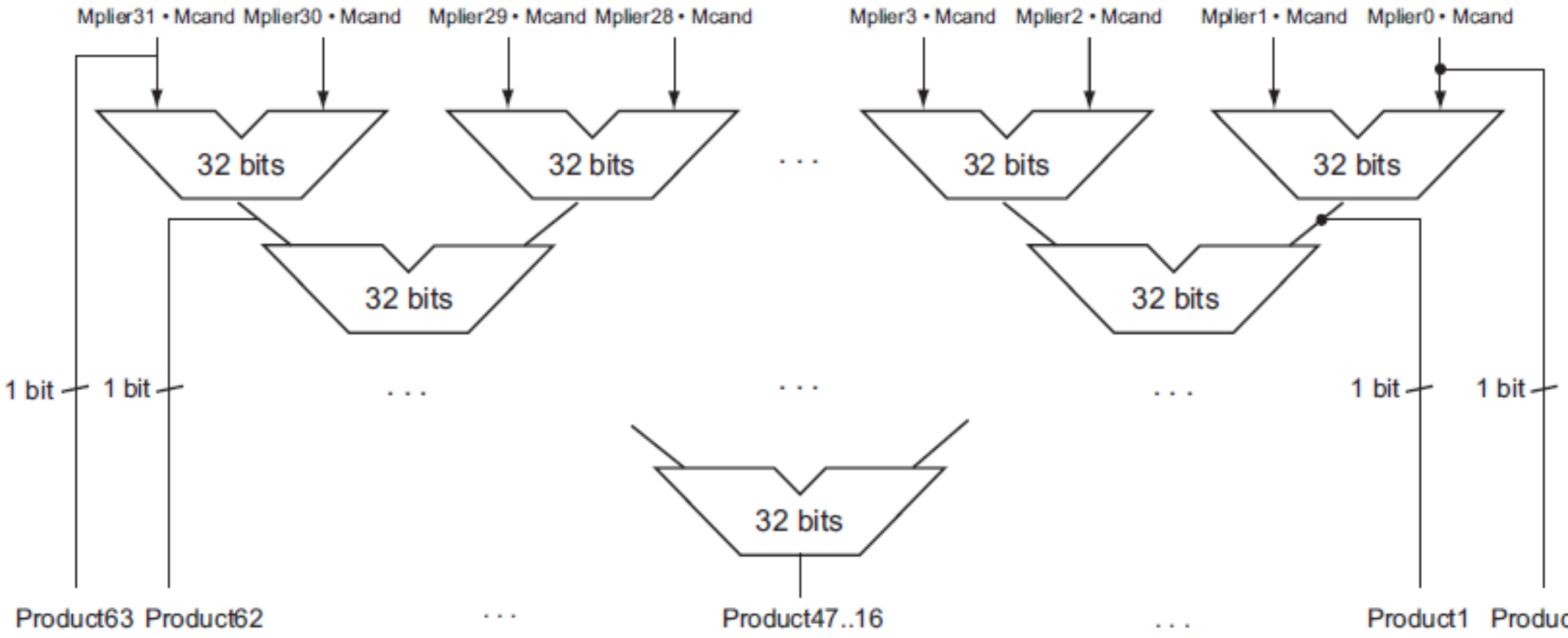
- The Multiplicand register and ALU have been reduced to 32 bits.
- The product is shifted right.
- The separate Multiplier register also not there.
- The multiplier is placed instead in the right half of the Product register, which has grown by one bit to 65 bits to hold the carry-out of the adder.
- Multiplicand not shifted left.

Signed Multiplication:

- First convert the multiplier and multiplicand to positive numbers
- Remember their original signs.
- Run algorithms for 31 iterations leaving the signs
- Negate the product only if the original signs disagree

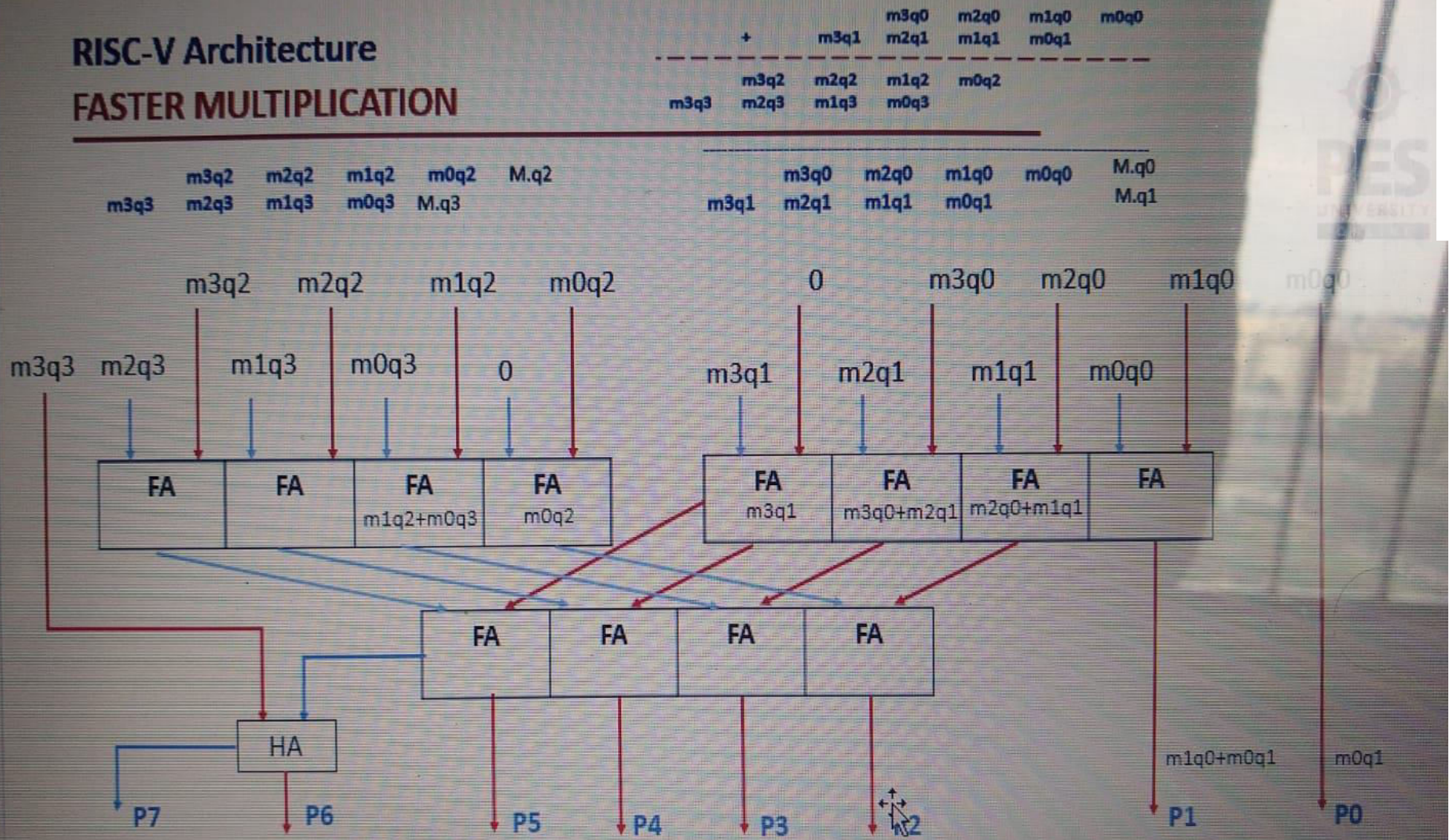
This algorithm will work for signed numbers also
The shifting steps would need to extend the sign of the product for signed numbers.
The lower word would have the 32-bit product.

Fast multiplication hardware: Use a single 32-bit adder 31 times, this hardware "unrolls the loop" to use 31 adders and then organizes them to minimize delay.


Mplier31 • Mcand
Mplier30 • Mcand
Mplier29 • Mcand
Mplier28 • Mcand
Mplier3 • Mcand
Mplier2 • Mcand
Mplier1 • Mcand
Mplier0 • Mcand
32 bits
32 bits
32 bits
32 bits
32 bits
32 bits
32 bits
1 bit
1 bit
1 bit
1 bit
Product63
Product62
Product47..16
Product1
Product0

RISC-V Architecture

FASTER MULTIPLICATION



Reference : Computer Architecture with RISC V - The Hardware/Software Interface: RISC-V Edition by David A. Patterson and John L. Hennessy

Faster Multiplication:

Use parallel tree structure - Instead of waiting for 32 add times, we wait just the $\log_2(32)$ or five 32-bit add times.

Use Carry save adders – pipelined architecture helps

Multiply in RISC-V:

- multiply (mul) - integer 32-bit product

- multiply high (mulh) - upper 32 bits of the 64-bit product if both operands are signed

- multiply high unsigned (mulhu) -- - upper 32 bits of the 64-bit product if both operands are unsigned

- multiply high signed-unsigned (mulhsu) -) upper 32 bits of the 64-bit product if one operand is signed and the other is unsigned

THANK YOU

Vanamala H R

Department of Electronics and Communication