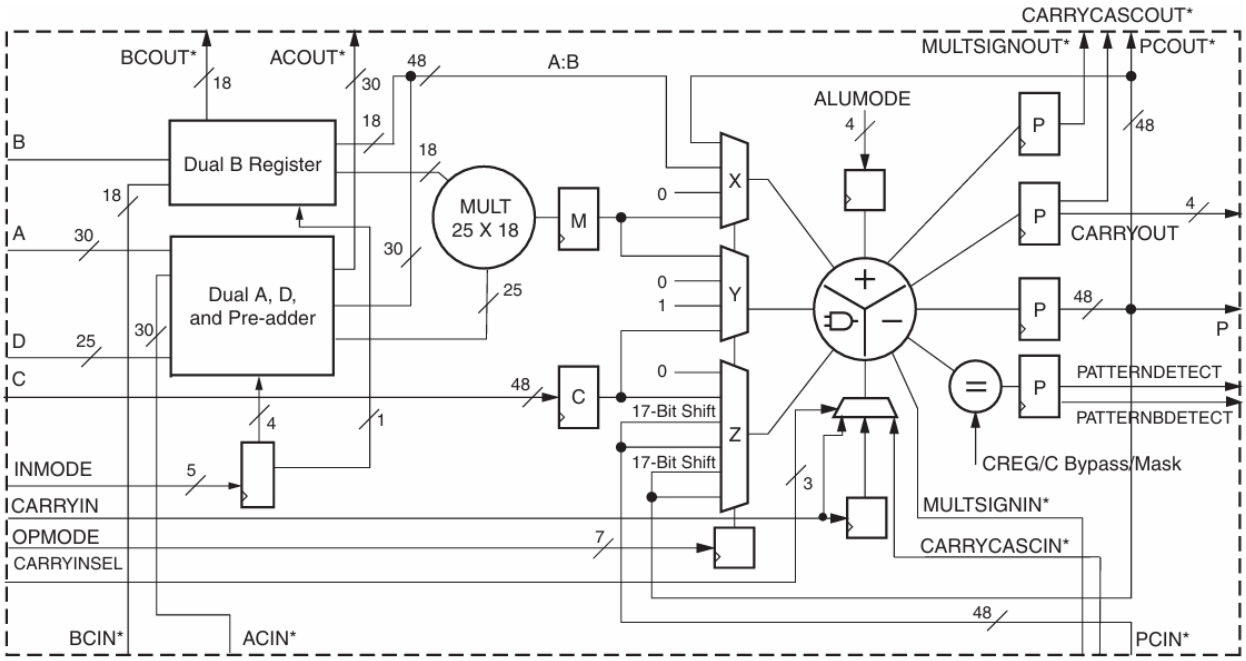
DSP48E Slice

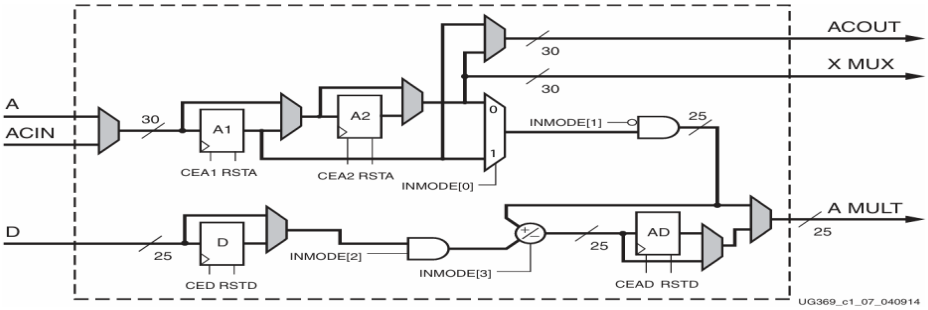


1-Artitecture of DSP48E Slice

A,B,C D Ports

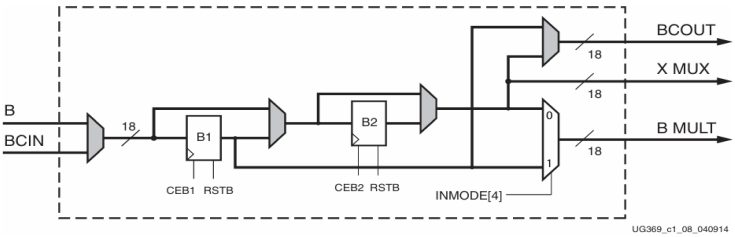
The input ports A, B, C, and D are 30, 18, 48, and 25 bits, respectively.

Dual A, D and Pre-adder



This block determines the input of 25x18 multiplier(pre-adder feature) and A:B(48 bits) input of X multiplexer. As it can be seen at figure, inmode register is a control register that control what is A MULT.

Dual B Register



This block determines, the input of 25x18 multiplier(pre-adder feature) and A:B(48 bits) input of X multiplexer.

Mult 25x18

The multiplier takes 18-bit and 25-bit two's complement inputs and produces a 43-bit result. This result is fed to the three-input adder by sign-stretching to 48-bit in multiplexers X and Y. Thus, the adder becomes a two-input adder during the multiplication operation.

C Port

The 48-bit C port serves as a general input for the Y and Z multiplexers, which execute addition, subtraction, three-input add/subtract, and logic operations. Input C is also linked to the pattern detector for use in rounding function applications.

CARRYINSEL Port Logic

The carry inputs X, Y and Z are determined before they reach the multiplexers and are independent of OPMODE. 8 different inputs can be selected with CARRYINSEL: CARRYIN from general logic, CARRYCASCIN from the neighboring DSP slice, its own CARRYCASCOUT feedback, A[24] XNOR B[17] to round the multiplier output, the MSB of the output or daisy-chained input P (P[47] or PCIN[47]), and their inverted forms.

X, Y, and Z multiplexer

The X Multiplexer (X MUX) is used to route the first operand to the adder/subtractor. In multiplier mode, the first 48-bit signal from the 25x18 bit multiplier result carries the extended portion. In non-multiplier mode, the A:B (combined 25-bit A and 18-bit B inputs) 48-bit C register can select the P output of the previous DSP slice or the PCIN input from another DSP slice. It also provides one of the operands for bitwise operations (AND, XOR, etc.) when the logic unit is active.

Y Multiplexer directs the second operand to the adder/subtractor. In multiplier mode, the second 48-bit sign carries the extended part, which completes the 43-bit multiplier result. In non-multiplier mode, it can select inputs such as C, P or PCIN for arithmetic operations, while for logical operations, it provides constants that are all 0 or 1, facilitating operations such as XOR. In addition, it can be used in carry management, providing CIN or constants as the carry input for arithmetic operations.

The Z Multiplexer (Z MUX) enables complex operations by directing the third operand to the adder/subtractor. In multiplier mode, it performs multiply-accumulate operations (MAC), usually by selecting the C register or the previous result P as the accumulator value. In this case, the output is calculated as Z ± (Multiplier Result + CIN). In non-multiplier mode, it provides the second operand for logical operations, while C, P or PCIN are used for addition or subtraction in arithmetic operations. Depending on the ALUMODE control signal, the Z value can be subtracted or its two's complement can be taken.

OPMODE Port Logic

OPMODE is 7 bit port, which controls which inputs will be connected to the X, Y and Z multiplexers. The inputs and outputs of these three multiplexers are configured via OPMODE bits. The OPMODE bits determine which inputs are connected to multiplexers X, Y, and Z. For example, when performing a MAC operation, the multiplier output is directed to X and Y, and the accumulator is directed to Z. Different OPMODE combinations can be used to perform operations such as (X + Y) + Z or Z - (X + Y). In addition, the ALUMODE control bit determines whether the Z operand is added or subtracted.

Add/Sub/Logic Unit

The second-stage adder/subtractor takes three 48-bit two's complement operands to produce a 48-bit result. It works when the multiplier is disabled (USE_MULT=NONE) and the appropriate OPMODE is set. In SIMD mode, it supports dual 24-bit or quad 12-bit SIMD operations with CARRYOUT bits. Additionally, bitwise logical operations on two 48-bit numbers can be performed dynamically with the ALUMODE control signals.

ALUMODE Port Logic
Alumode is 4-bit port, which control second stage of add/sub/logic unit.

P Port
The DSP48E1 slice has a 48-bit P output, which can be connected to the adjacent DSP48E1 slice as a PCIN input via the PCOUT bus. This structure is used to provide data flow between adjacent slices.

2. Advantages of DSP48E1 Slice

High Performance:
Optimized for 25x18-bit multiplication and 48-bit accumulation operations.
Clock frequencies can exceed 500 MHz, making it ideal for high-speed DSP applications.

Flexibility:
Supports various operations such as multiplication, addition, subtraction, logical operations, shift (barrel shift), and pattern detection.
With SIMD (Single Instruction, Multiple Data) mode, the 48-bit ALU can be divided into 2x24-bit or 4x12-bit, facilitating parallel operations.

Low Power Consumption:
Compared to FPGA fabric, DSP slices consume less power. Especially when DSP slices are used for multiplication and addition operations, power efficiency is increased.

Cascading Feature:
DSP48E1 slices can be connected to each other via cascading. This is ideal for large filters or wide bit-width mathematical operations.

Pattern Detection and Rounding Support:
Pattern detection logic supports operations such as overflow, underflow, and symmetric rounding.

3. Why is the DSP48E1 Slice Used?

Digital Signal Processing (DSP):

It is ideal for signal processing algorithms such as FIR (Finite Impulse Response) and IIR (Infinite Impulse Response) filters, fast Fourier transforms (FFT), convolution.

High-Speed Mathematical Operations:

It is used for high-speed mathematical operations such as matrix multiplication, complex number operations, polynomial calculations.

Data Path Operations:

Optimized for wide-bit-width data path operations (e.g., 48-bit addition/subtraction).

Control and Counting Operations:

The DSP48E1 slice can also be used for high-speed counter and control logic applications.

4. Design Recommendations for DSP48E1 Slice

Using Pipeline:

Use pipeline to achieve full performance in DSP48E1 slice. A 3-stage pipeline is recommended for multiplication operations and a 2-stage pipeline is recommended for addition operations.
Increase performance by using M register, especially for multiplication operations.

Reduce Power Consumption:
Reduce static power consumption by disabling unused DSP slices.

For small bit-width multiplication operations, place operands in MSBs and reset LSBs.

Use cascading paths to establish direct connections between DSP slices instead of fabric routing.

Cascading and Adder Tree Structures:
Use adder cascade structure for large filters or addition operations. This reduces power consumption and improves performance.

Adder tree structure may consume more resources but can give faster results. Choose the structure that suits your design.

Time Multiplexing:
In low-sample-rate applications, use a single DSP slice for multiple channels. This reduces resource usage and power consumption.

Using CLB for Small Operations:
Use CLB (Configurable Logic Block) for small multiplication operations (e.g., 4x4) and small bit-width adders. This reserves DSP slices for more complex operations.

Pattern Detection and Rounding:
Optimize overflow, underflow, and rounding operations using pattern detection logic. This is especially useful for filtering and accumulation operations.

Coefficient Storage:
Use CLB SRL16/SRL32 or block RAM to store filter coefficients. This increases the data processing capacity of DSP slices.
