IEEE ASUSB

Digital Design Team

NOV.2025

Filtration Task

Generic Carry Look Ahead Adder

Due date:

SUNDAY 13/11/2025

Requirements:

RTL: the Verilog code implements the assigned question and comment your design well.

Simulation: create a simple Testbench for the design and run simulation attach screenshots to show the Transcript output and waveform

FPGA: run synthesis and Implementation using VIVADO Toolchain targeting VCU118 board and report the Utilization and critical path delay calculation

Parametrized RTL: write generic code can be used to generate N-bit adder (where N is Parameter/Generic) of type similar to the Type in the Assigned Question

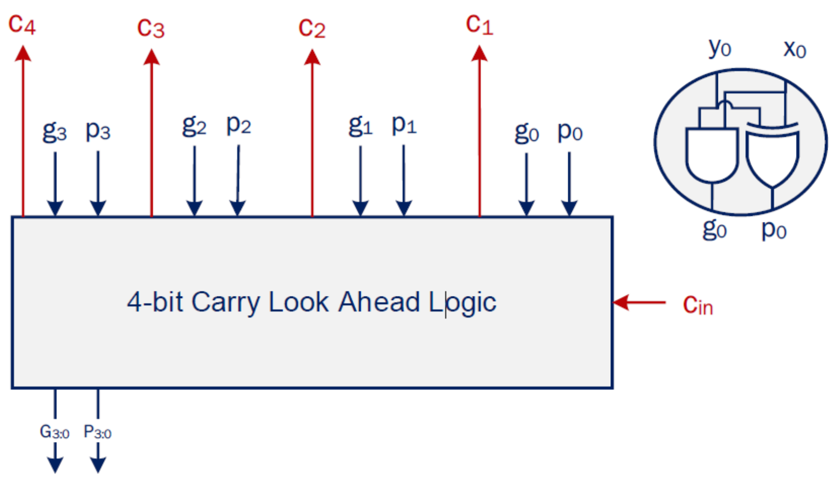
Report: report the main flaws (if any) in your design and explain the limitations in your generic implementation.

Specs:

You are asked to design a 64-bit hierarchical Carry look ahead adder using 4-bit building unit through the following the steps below and attach all the required captures and codes.

• **Step 1: 4-bit CLA (carry network) :**

Definition: design a 4-bit Carry look ahead carry network which use generate propagate signals in addition to carry in and compute all the carry signals C1 to C4. The circuit below describes how to compute generate, propagate signals from x and y inputs

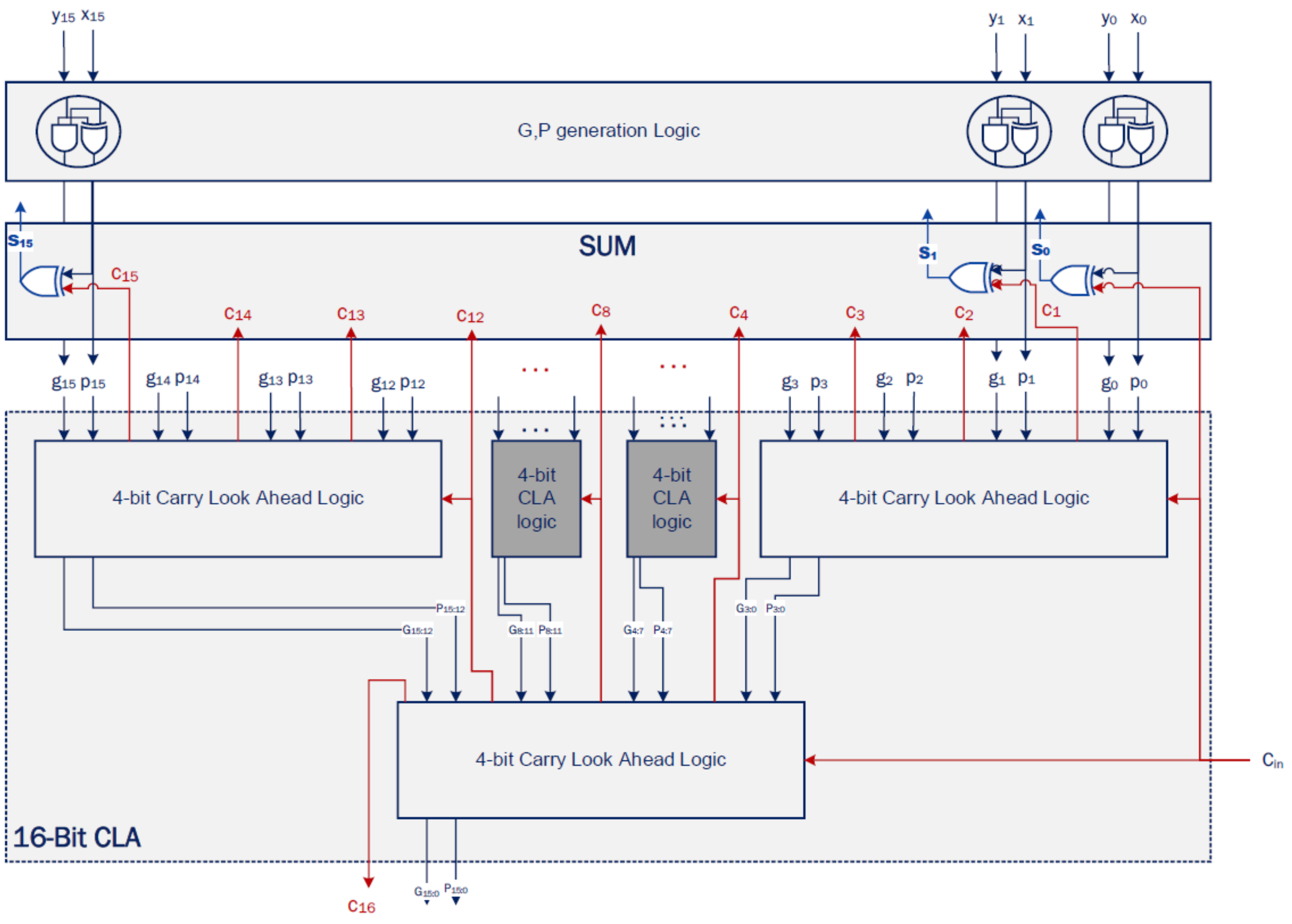


- **Inputs** { cin, g0, p0, g1, p1, g2, p2, g3, p3 }

- **Outputs** {*c1*, *c2*, *c3*, *c4*, *G0:3*, *P0:3* }

• Step 2: 64-bit hierarchical Carry look ahead adder using 4-bit CLA logic as building block

Definition: Hierarchical Carry Look Ahead Adder uses smaller size of carry look Ahead logic nested to compute all the carry locations and avoid the growth in the gate sizes and number of inputs in the figure below a full description how to build 16-bit adder using 2 levels of 4-bit carry look ahead module to build carry network then use the carry signals with propagate signal to compute the sum trace back the schematic below and convince yourself about how the operation done then use the same idea one more level to build 64-bit adder. Nothing will change in *SUM* or *G,P generation logic* only the carry network will change by adding one more level



- Inputs {*A*, *B*, *Cin*} **(***A*,*B* **are 64-bits inputs and** *Cin* **is 1-bit)**

- Outputs {*Sum*, *Cout*} **(***sum* **is 64-bits output and** *Cout* **is 1-bit)**

References:

- Computer arithmetic _ algorithms and hardware designs -- Parhami, Behrooz -- Oxford series in electrical and computer engineering, 2nd (Chapter 6)

- UCSB ECE 252B, Spring 2020, Lecture 4: Carry-Lookahead Adders