

# Ripple 3-bit Adder

## Hierarchical Design, Adders & Registers

### 1. Introduction

This program introduces sequential circuits such as flip-flops and registers, create an adder, and use it in combination with registers to implement a simple ALU. As a side effect of building bigger things from smaller things, the concept and use of hierarchy in design are illustrated.

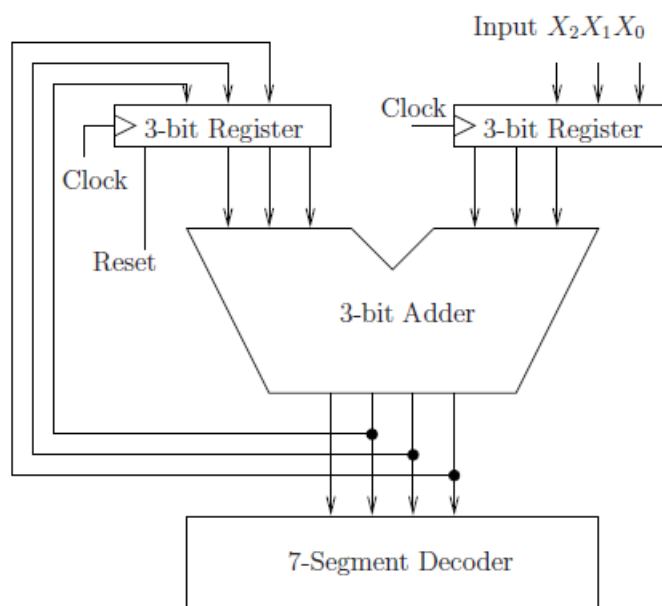


Figure 1: Adder Unit Schematic

In the file folder, the files are for the presentation using the circuit board of Cyclone V FPGA (5CSEMA5F31C6) device. You can use other board for demonstration, but you need to change the device connection in Quartus Prime.

### 2. Components

a) hexadecimal 7-segment display driver

Essentially, for the decimal numbers 0 to 15 (corresponding to 4 bit binary numbers 0000 to 1111 as input), a component of the display driver will display the corresponding hexadecimal digits 0 to F.

A file called "hex7seg.v" is used for this circuit.

b) The Verilog file "alu.v" is to build and simulate an edge triggered D type flip flop

c) The Verilog file “register.v” is to build and simulate the register.

d) The Verilog file “adder.v” is to build and simulate a 3-bit D register with an asynchronous reset signal.

### **3.Demonstration**

Demonstration of this circuit can be done by compiling it and downloading it to the DE1-SoC board using Quartus Prime.

Use toggle switches SW[9], SW[8], and SW[7] for inputs X2,X1,X0 respectively. Use one HEX led as output to display the alu result. Connect the reset signal to pushbutton KEY[1], and the clock to pushbutton KEY[0] (active low pushbutton) to generate reset or clock pulses. (The instruction of the set up of the switches and buttons is in the DE1-SoC board refer user manual)