

Lab 8 – Tally Unit

CS 2052 Computer Architecture

Dept. of Computer Science and Engineering, University of Moratuwa

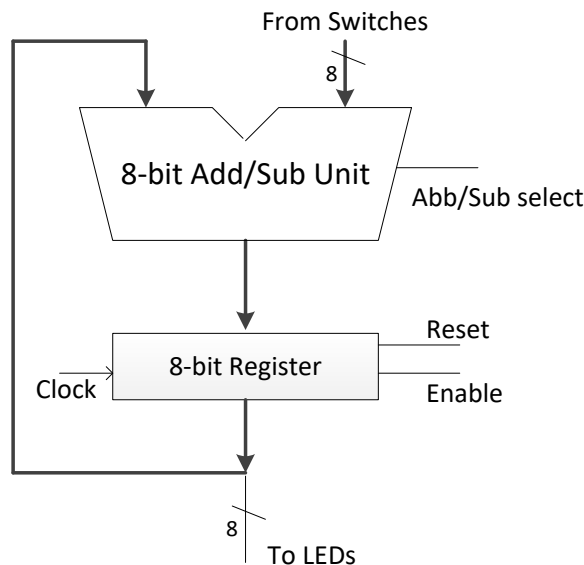
Learning Outcomes

In this lab, we will design an 8-bit add/subtract unit attached to a register. After completing the lab, you will be able to:

- design and develop a 8-bit arithmetic unit that can add and subtract integers
- develop and use a bus
- verify their functionality via simulation and on the development board

Introduction

We will design a simple tally unit that is capable of adding or subtracting numbers from a value stored in a register. Block diagram of the tally unit is given below.



To build this circuit, we need a couple of extensions for the components that we have already built. First, we need to extend our 4-bit Ripple Carry Addder (RCA) to be 8-bits and support both addition and subtraction. Second, we need to build an 8-bit register with both clear and enable inputs. We will also use 8-bit data buses in this design.

Building the Circuits

Step 1: Building 8-bit Add/Sub Unit.

Import RCA symbol schematics from Lab 5. Extend it to support 8-bits and subtraction operation by representing the numbers as 2's complement values. Name the schematic file as **Add_Sub**.

Label Add/Sub select pin as **AS**.

Use 8-bit data busses to connect to the 2 data inputs to the add/subtract circuit and data output. Label the buses as **A(7:0)**, **B(7:0)**, and **S(7:0)**. See "A Tutorial on Using Bus in Xilinx ISE" (available on the LMS and http://cseweb.ucsd.edu/classes/su06/cse140L/tutorial_bus.pdf) to find out how to create a bus in Xilinx ISE. Create a symbol name **Add_Sub**.

Step 2: Building 8-bit Register.

Name the schematic file as **Reg_8**. Build an 8-bit register with Enable, Clock, and Clear inputs. Use D Flip Flops, each with a Clear pin (look for a component name FDC), to build this register. Use data busses to connect data inputs and outputs. Label the busses as **D(7:0)** and **Q(7:0)**. Label other pins as **EN**, **CLK**, and **CLR**. Create a symbol name **Reg_8**.

Step 3: Building Tally Circuit.

Using **Reg_8** and **Add_Sub** symbols build the above tally circuit. Connect the switches **SW0-SW7** as one of the inputs to **Add_Sub**. Use pushbutton **BTN0** as the **CLK** input. Use **BTN1** as the **CLR** input. Use **BTN3** as the **AS** input (circuit should add the values when **BTN3** is not pressed). Direct the outputs to LEDs **LD0-LD7**. We do not need to use the **EN** input in this circuit (it will be useful in next lab). Therefore, hardware **EN** to logical **1**.

Step 4: Testing the Circuit.

Simulate the circuit and make sure it is working correctly.

Generate the programming file (i.e., bitstream) and load it to the BASYS2 board.

Change the switches on the BASYS2 and verify the functionality of your circuit.

Demonstrate the circuit to the instructor and get the Lab Completion Log signed.

No lab report is due for this lab.

Prepared By

- Dilum Bandara, PhD – Mar 27, 2014.