Two's Complement Adder/Subtractor
Lab L03

Introduction

Computers are usually designed to perform indirect subtraction instead of direct subtraction. Adding **-B** to **A** is equivalent to subtracting **B** from **A**, so the ability to add negative numbers implies the ability to do subtraction. Addition is relatively simple with two's complement numbers because two's complement numbers can be added by ordinary binary addition, ignoring any carries beyond the most significant bit. Some examples of decimal addition and the corresponding 4-bit two's-complement addition confirm this:

4 - 6 = -2		**7 - 5 = 2**	
+4	**0100**	**+7**	**0111**
+(-6)	**1010**	**+ (-5)**	**1011**
-2	**1110**	**+ 2**	**0010**

+

(Observe that the output carry in the second addition, that is, (+7 + (-5)), is ignored.) Using an adder to perform subtraction thus removes the requirement for a subtractor circuit. Practically all hand-held calculators and computers utilize the indirect subtraction by addition method to reduce their circuitry.

A full-adder is a logic circuit that adds three 1-bit binary numbers x, y and z to form a 2-bit result consisting of a sum bit and a carry bit. For example, if **x = y = z =1**, the full adder should produce **Carry = 1**, **Sum =1**, corresponding to the binary number 11, that is 3. The truth table of a full-adder is listed in Figure 3a. The eight rows under the inputs designate all possible combinations of the three inputs **x, y** and **z**. The two outputs **Sum** and **Carry** are determined from the arithmetic sum of the input bits. The sum output is equal to 1 when only one input is equal to 1 or when all three inputs are equal to 1. The carry output is equal to 1 if two or three inputs are equal to 1.

Inputs Outputs

x	y	z	Carry	Sum
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1

Figure 3a: Full adder truth table

Figure 3b shows a possible gate-level realization of a full adder using AND, OR and XOR gates.

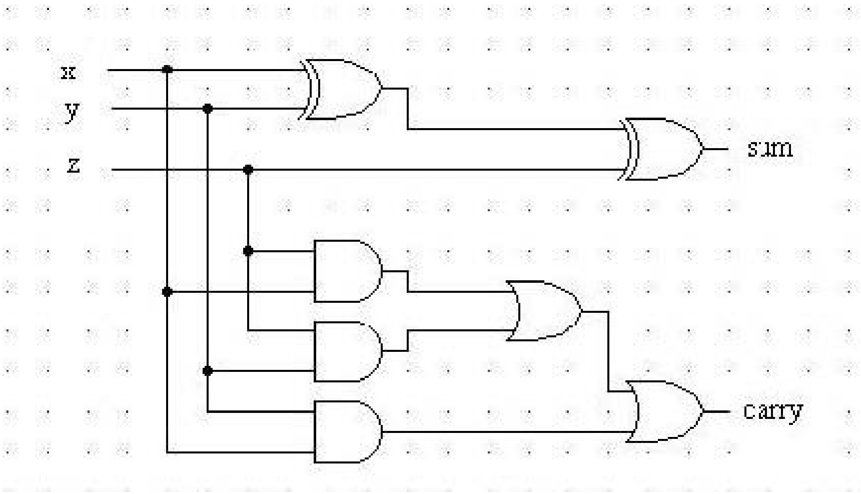


Figure 3b: Full adder gate-level circuit

A full adder is a versatile circuit; for instance, it can be used to build multibit adders and/or subtractors. In this lab, you are first going to enter the circuit of Figure 3b in Digital Works and convert it into a black box with 3 inputs and 2 outputs, as shown in Figure 3c.

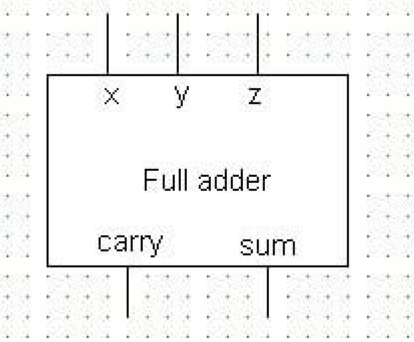


Figure 3c: Full adder block symbol

The black box circuit is called a macro in Digital Works; it can be used as a building block in the construction of more complex circuits. The appendix of this lab handout contains a description of how to create/use a macro in Digital Works. For now, let's look at the overall problem that you

Design

Using four of the full adder circuits of Figure 3c, design a circuit that adds two 4-bit numbers **A** and **B**, that is, a 4-bit adder. Consider operand **A** to be made up of the bits **a3 a2 a1 a0** and operand **B** to be made up of the bits **b3 b2 b1 b0**. Consider the sum of **A** and **B** to be made up of the bits: **3 s2 s1 s0**. Enter your circuit in Digital Works and verify the correctness of your design by simulating the operation of the 4-bit adder. For example, if **A** is 0011 (decimal 3) and **B** is 0001 (decimal 1) then the resulting sum bits from your adder circuit should be 0100.

Once you figure out how to build a 4-bit adder your next task is to implement addition as well as subtraction in one common circuit. Modify your 4-bit adder circuit by introducing a mode input **M**. When **M = 0**, the modified circuit should behave as an adder (that is, perform **A + B**), and when **M = 1**, the modified circuit should behave as a subtractor (that is, perform **A - B**). Modifying the 4bit adder circuit to perform two's complement subtraction (as well as addition) merely requires connecting suitable two-input logic gates to the full adders' inputs and utilizing **all** three inputs of the full adder that adds the two operand bits **a0** and **b0**.

Submitting Results

All macros are to be submitted with your assignment. Please make sure that each macro includes your name, datte and assignment number. Also that they are named correctly ie: LastnameF_L03x (Where Lastname is your last name, F is your first initial and x is a unique letter [a-z] that you used to distinguish each macro.

Lab L03: Appendix

Creating a macro

Once a design of a useful function is known, such as the full adder circuit of Figure 3d, Digital Works enables you to place it in a library for future use. Turning circuits into reusable black boxes is analogous to the use of subroutines in a high-level language.

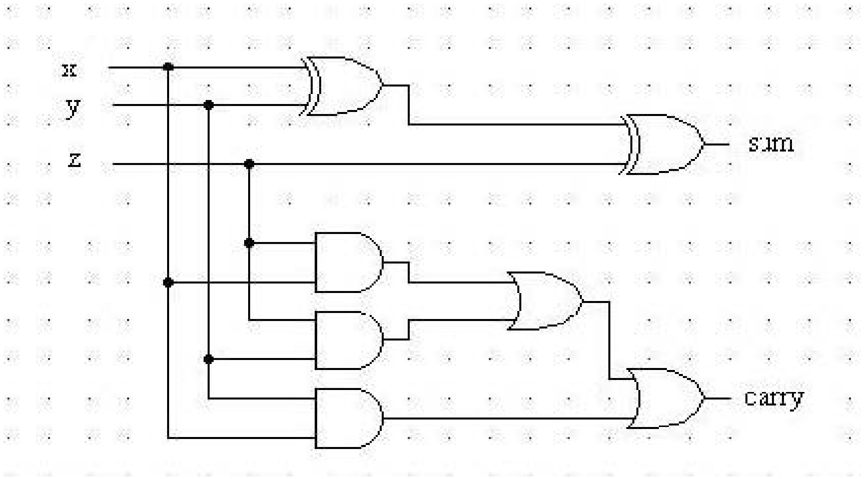
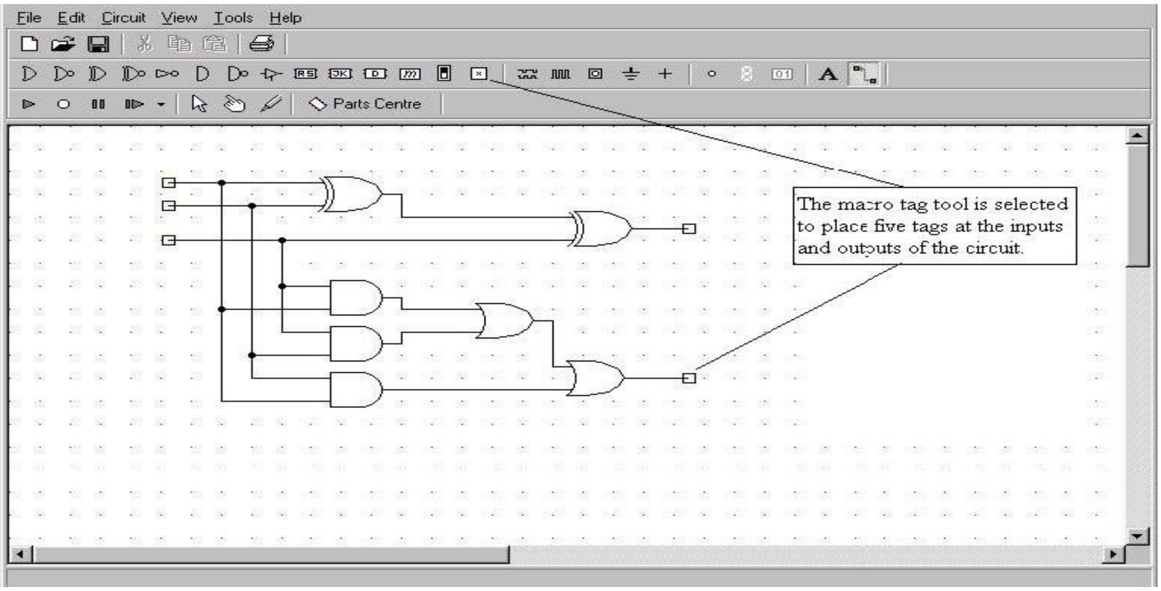


Figure 3d: Full adder circuit

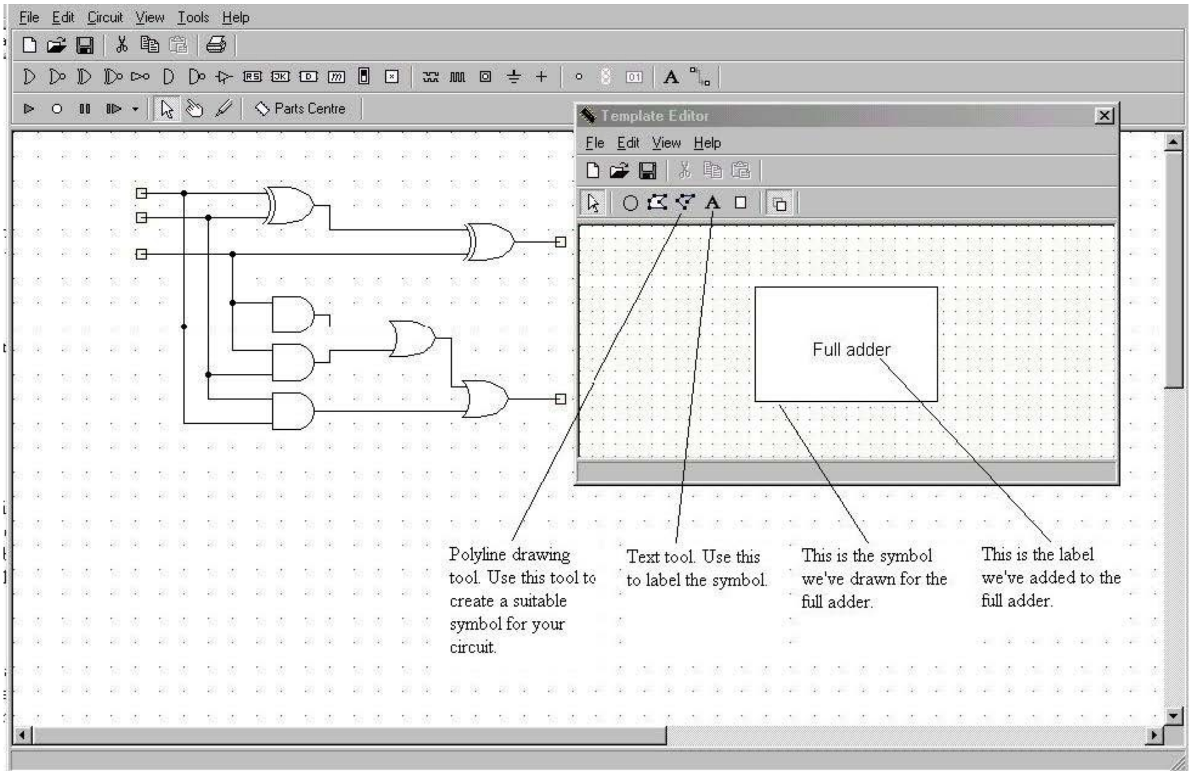
Let's take the circuit of Figure 3d and convert it into a black box with 5 terminals: three inputs **x, y**and **z**, and two outputs **sum** and **carry**. The first step is to place macro tags at the circuit's inputs and outputs. A macro tag can be wired up to the rest of the circuit exactly like an input or output device. You left click on the macro tag icon to select it and then move the cursor to the place on the workspace you wish to insert the macro tag (that is, the input or output). Then you wire the macro tag to the appropriate input or output point of the circuit. Note that you can't apply a macro tag to the input or output of a gate directly --- you have to connect it to an input or output by a wire. See Figure 3e.



The macro tag tool is selected to place five tags at the inputs and outputs of the circuit.

Figure 3e: Placing tags in the full adder circuit

The next step is to create the black box itself. This is a slightly involved procedure. Place the cursor over one of the macro tags in the original circuit (that is, Figure 3e) and right click to pull down a menu. Select **Template Editor** from the menu with a left-click. A new window called **Template Editor** appears (Figure 3f). You create a black box representation of the circuit in this window. We have to use the simple polyline drawing tool provided by Digital Works create a suitable shape for the representation of the full adder. You just click on this tool in the **Template Editor** window and draw the shape by clicking in the workspace at the points you wish to draw a line. You exit the drawing mode by double clicking. You can also add text to the drawing by using the text tool. Figure 3f shows the shape we've drawn for the full adder and the label we have given it.



Polyline drawing tool. Use this tool to create a suitable symbol for your circuit.

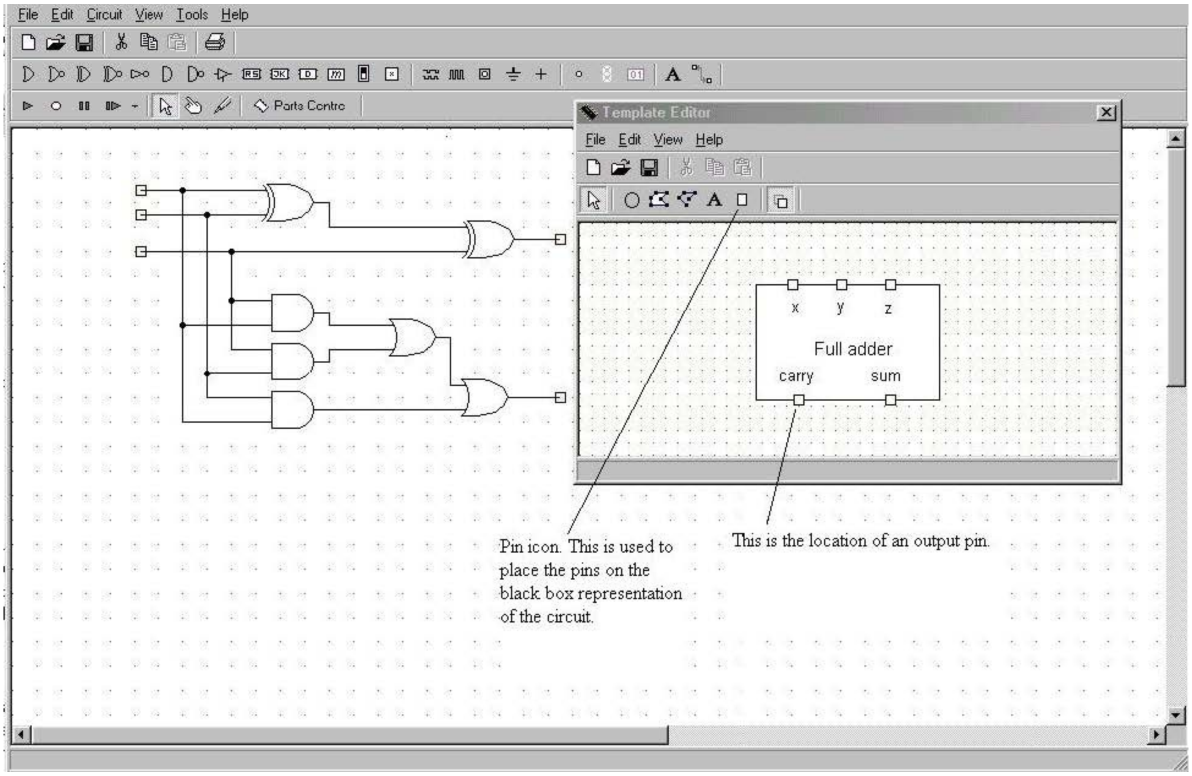
Text tool. Use this to label the symbol.

This is the symbol we've drawn for the full adder.

This is the label we've added to the full adder.

Figure 3f: Drawing a symbol for the full adder

The next step is to add pins to the black box in the **Template Editor** window and associate them with the macro tags in the original circuit. For each macro tag in the original circuit, you will need to insert a pin in the black box representation. To place a pin click on the pin icon in the **Template Editor** and then left-click in the workspace at the point you wish to locate this pin. Once you add all the pins use the text tool to add labels to the five pins (see Figure 3g). Close the **Template Editor** to bring you back to the circuit diagram.



Pin icon. This is used to place the pins on the black box representation of the circuit.

This is the location of an output pin.

Figure 3g: Adding pins and labels

For each macro tag in the original circuit, repeat the following operations:

- Right-click the macro tag to enter the template editor.
- In the template editor, select the corresponding pin. You then right click on this pin and select **Associate with Tag** to associate the pin with the macro tag in the circuit diagram.
- Close the template editor.

Digital Works automatically numbers the tags in the circuit diagram as you associate them with pins. When you are finished assigning pins to macro tags, left-click the **Tools** command in the top line, then click **Name Macro**. Enter "**1-bit full adder**" into the text box and click **OK**. Finally, save your original circuit. This also saves its black box representation. Note that the template editor also has a save function; this save function does not save the circuit. You will need to save the circuit in a special folder called **Macros**, a subfolder of the **Parts Centre** folder. Enter **LAB2_FA_XX** (where XX are your initials) as the name of your file. (Note:if you do not have the permission to save in the above folder, check with the TA to where you should save the Macro.)

Using a macro

The macro you have created for a 1-bit full adder and saved can be used like other circuit elements. Start a new circuit in Digital Works and begin with an empty work area. Click on the **Part Centre** icon in the bottom toolbar (Figure 3h). Select the appropriate macro from the pull-down menu that appears by left-clicking the embed macro icon and drag it wherever you want to place it in the workspace. If you saved your macro in a different drive/folder, make sure to click on the **Change Drive** button, shown in Figure 3h, to change the macros drive.

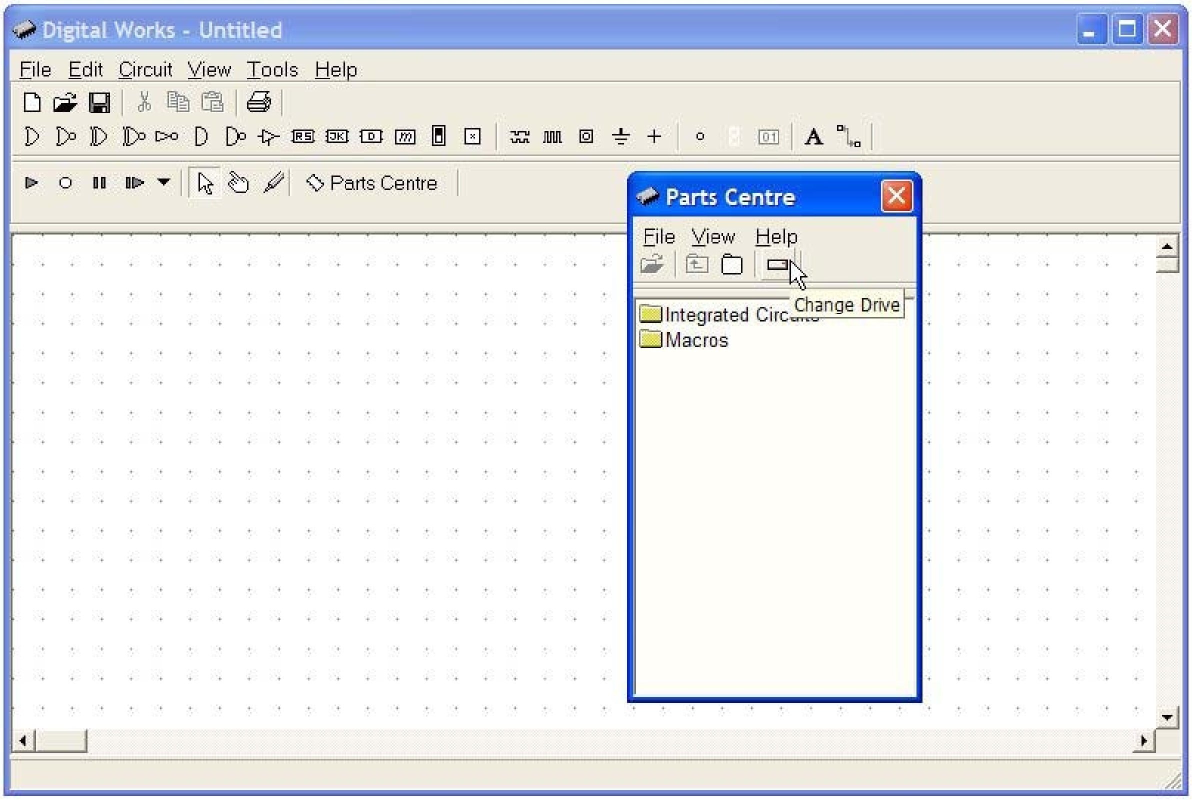


Figure 3h: Embedding a macro in a circuit