Introduction to Digital Logic

Tom Kelliher, CS 220

1 Administrivia

Today's Objectives

1. Achieve a basic understanding of combinational and sequential digital logic:
   (a) AND, OR, NOT, etc. gates.
   (b) Basic combinational circuits: full- and half-adders, decoder, multiplexer, etc.
   (c) Flip-flops, registers, and counters.
   (d) Basic implementation of buses.

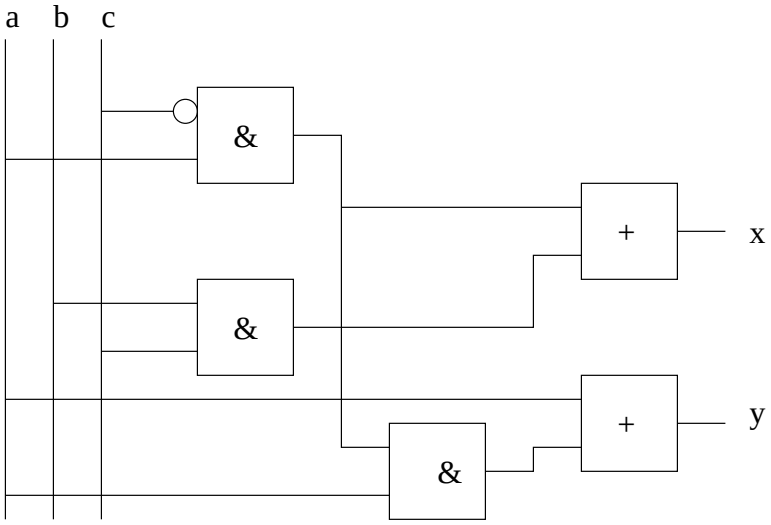
Next Up

Read 7.2.1.

1. Understand the capabilities of the register-to-register data path

2 Warm-Up

1. The Boolean equation for output `x` is



(a) $ab\overline{c} + a$

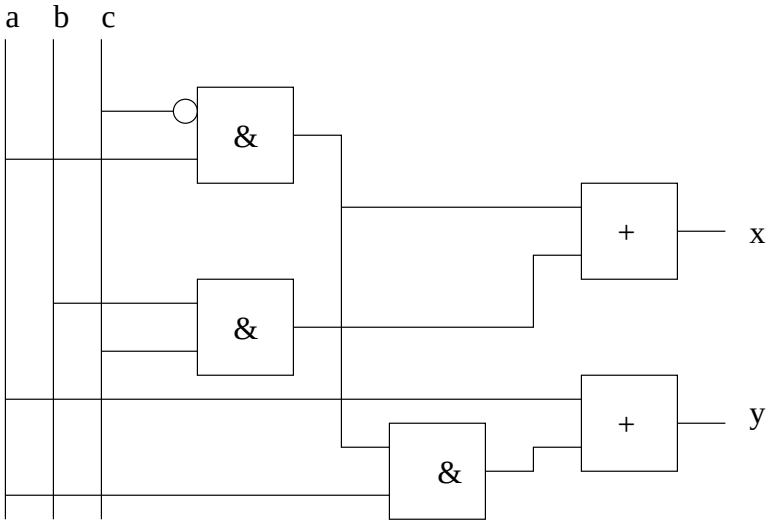
(b) $a\overline{c}$

(c) $ac + bc$

(d) $a\overline{c} + bc$

(e) None of the above.

2. The Boolean equation for output y is



(a) $ab\overline{c} + a$

(b) $a\overline{c}$

(c) $ac + bc$

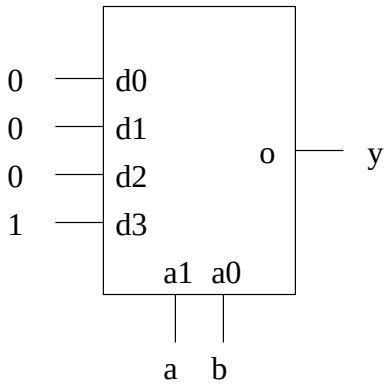
(d) $a\overline{c} + bc$

(e) None of the above.

3. The Boolean equation $ab\bar{c} + a$ simplifies to a.

   True/False

4. The Boolean function computed by this mux is



(a) $\overline{a}$

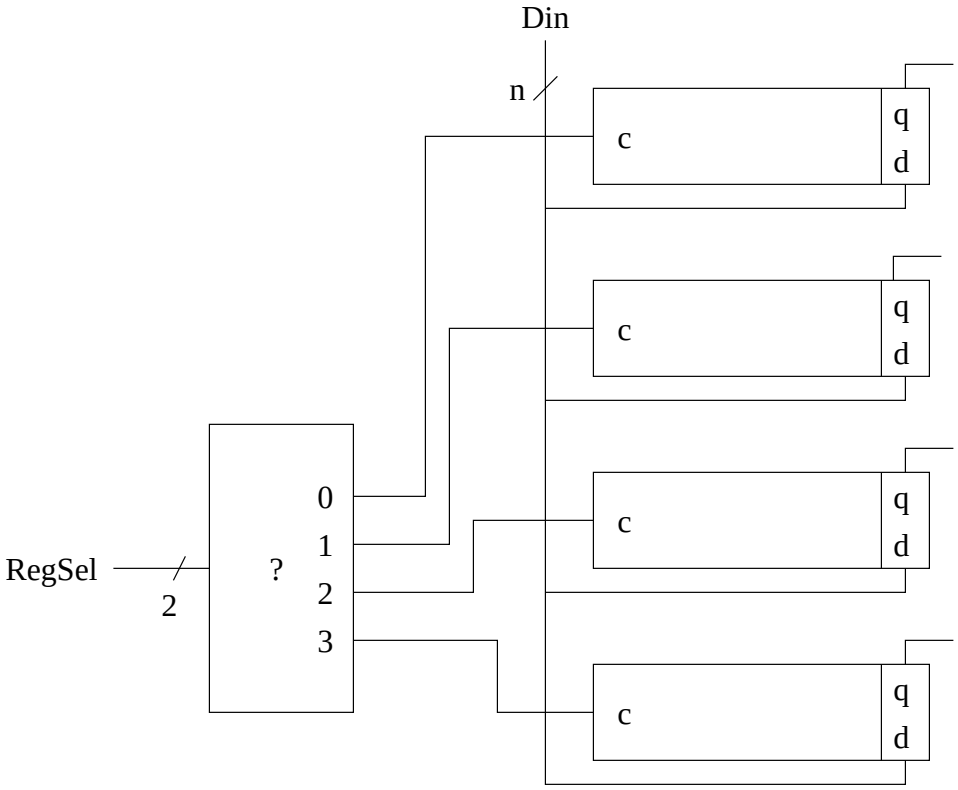
(b) $a + b$

(c) ab

(d) $\overline{ab}$

(e) a `XOR` b

5. The mystery combinational logic element, labeled with `?`, in this figure is a



(a) Decoder

(b) Mux

(c) ALU

(d) If I told you, I'd have to permanently "disable" you.

6. What digital logic element, or elements, should be used to read one of many registers onto a single bus?

   (a) Muxes

   (b) A decoder in combination with muxes.

   (c) A decoder in combination with tri-state buffers.

   (d) (a) and (b)

   (e) (a) and (c)

3 Problem Set 11.0

1. Complete the one-bit ALU on pg. 94 of the textbook by supplementing it with an add/subtract capability.

2. Using a bank of 16 32-bit registers and a 32-bit ALU, design a data path that could be used to execute ARM instructions such as

```
add r0, r1, r2
and r3, r4, r5
orr r6, r7, r8
```