

CSSE2010/CSSE7201
Lecture 7

Sequential Circuits 2

Counters

School of Information Technology and Electrical Engineering
The University of Queensland

Today

- Admin
 - Quiz 3 –due on Friday 4PM AEST this week
 - Check my grades (you need to click on your mark) to get feedback on quizzes 1 and 2
 - Quiz 2 – attempts 403, average 8.3/9, median 8.3/9
 - Lab Preparation Reminder – lab 6 and lab 7
 - Some changes made to teaching outline and ECP section 5. Please take note.
- Shift register question
- More sequential circuits
 - Counters

From ECP – Section 5

Assignment 1 (Digital Logic Design)

Type: Laboratory

Learning Objectives Assessed: 1, 2, 3, 4

Due Date: 03 Sep 21 - 06 Sep 21

Weight: 20%

Task Description:

Assignment 1 will have two parts.

- Part 1: this will be a timed (1-hour) Blackboard quiz with 25 multiple choice questions which should be completed in a single attempt. **Due date for part 1 is 3 Sep 2021 4:00 PM AEST.**
- Part 2: students will undertake an **individual** lab task to design and simulate a digital system for a given set of specifications. The scope and complexity of the task will be based on lab activities from week 1-5 (inclusive) and students are required to submit their design workings and simulation files via Blackboard. Any student can be called for a subsequent oral assessment if signs of plagiarism are detected in their submission. **Due date for part 2 is 6 Sep 2021 4:00PM AEST.**

Criteria & Marking:

UQ Students: Please access the profile from [Learn.UQ](#) or [mySI-net](#) to access marking criteria held in this profile.

Submission:

Submissions will be via Blackboard.

From Teaching Outline

| Wk | Day | Date | Lecture (date) | | Learning Lab | | Assessment due |
|----------------|---------|-----------|----------------|--|--------------|---|--|
| 1 | Mon-Tue | 26-27 Jul | 1 | Course Introduction and Binary numbers (26 Jul) | | No labs (P2 sessions) on Mon-Tue on week 1 | |
| | Wed-Fri | 28-30 Jul | 2 | Logic gates (28 Jul) | 1 | Signed number representations | |
| 2 | Mon-Tue | 2-3 Aug | 3 | Binary Arithmetic (2 Aug) | 2 | Logic Gates | Quiz 1 (4pm, 6 Aug) |
| | Wed-Fri | 4-6 Aug | 4 | Combinational Logic (4 Aug) | 3 | Binary Arithmetic | |
| 3 | Mon-Tue | 9-10 Aug | 5 | Flip flops (9 Aug) | 4 | Combinational Logic | Quiz 2 (4pm, 13 Aug) |
| | Wed-Fri | 12-13 Aug | 6 | Sequential Circuits 1 - Shift Registers (11 Aug) Pre-recorded lecture due to public holiday on 11/08/21 | 5 | Flip flops | |
| 4 | Mon-Tue | 16-17 Aug | 7 | Sequential Circuits 2 - Counters (16 Aug) | 6 | Sequential Circuits 1 | Quiz 3 (4pm, 20 Aug) |
| | Wed-Fri | 18-20 Aug | 8 | Sequential Circuits 3 - State Machines (18 Aug) | 7 | Sequential Circuits 2 | |
| 5 | Mon-Tue | 23-24 Aug | 9 | ALU and Memory (23 Aug) | 8 | Sequential Circuits 3 | Quiz 4 (4pm, 27 Aug) |
| | Wed-Fri | 25-27 Aug | 10A | Control Unit, Assignment 1 release (25 Aug) | | Cath-up labs, Assignment 1 release end of week | |
| 6 | Mon-Tue | 30-31 Aug | 10B | Review and Introduction to C language (30 Aug) | | Assignment 1 working time | Assignment 1 - part 1 due date (4pm, 3 Sep) |
| | Wed-Fri | 01-03 Sep | 11 | Introduction to AVR and Assembly Language (1 Sep) | | Part 1: 1-hour timed quiz Part 2: Digital logic design | |
| 7 | Mon-Tue | 06-07 Sep | 12 | Instruction Set Architecture 1 (06 Sep) | 9 | C Programming | Assignment 1 - part 2 submission (4pm, 6 Sep) Quiz 5 (4pm, 10 Sep) |
| | Wed-Fri | 08-10 Sep | 13 | Instruction Set Architecture 2 (08 Sep) | 10 | Atmel Assembly | |
| 8 | Mon-Tue | 13-14 Sep | 14 | C for AVR Microcontroller (13 Sep) | 11 | AVR Hardware Introduction | Quiz 6 (4pm, 17 Sep) |
| | Wed-Fri | 15-17 Sep | 15 | Flow Control and AVR Timers (15 Sep) | 12 | AVR C Programming | |
| 9 | Mon-Tue | 20-21 Sep | 16 | Pulse Width Modulation (PWM) (20 Sep) | 13 | AVR Timers | Quiz 7 (4pm, 24 Sep) |
| | Wed-Fri | 22-24 Sep | 17 | Interrupts (22 Sep) | 14 | AVR PWM | |
| Semester Break | | | | | | | |

Which of the following statements about the circuit below is true?

17% A. When A is 1, flip-flop values are shifted to the right

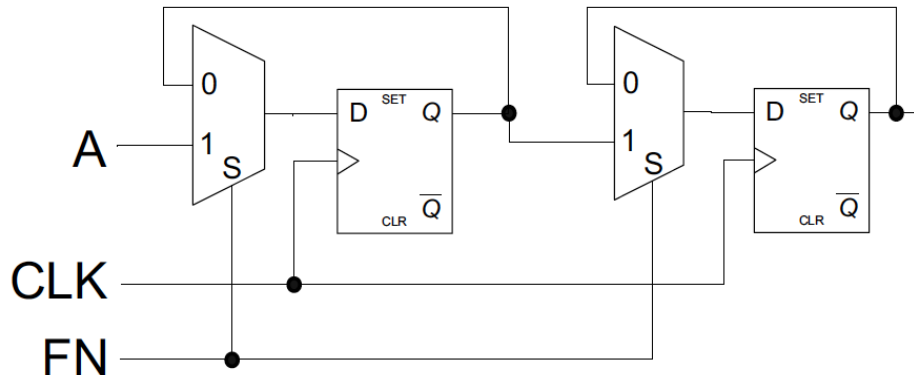
17% B. When A is 0, flip-flop values are shifted to the right

17% C. When A is 0, flip-flop values stay the same

17% D. When FN is 1, flip-flop values are shifted to the left

17% E. When FN is 0, flip-flop values stay the same

17% F. When FN is 0, flip-flop values are shifted to the left



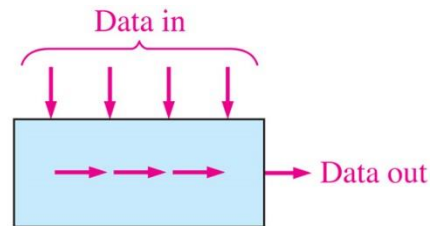
Shift Register Types – From Last Week



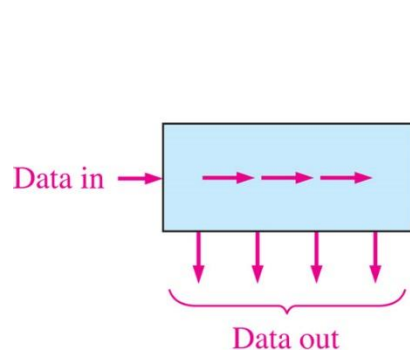
(a) Serial in/shift right/serial out



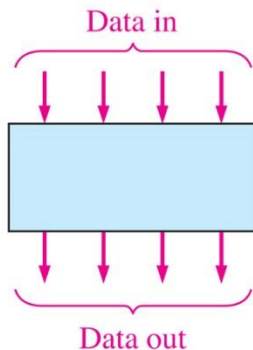
(b) Serial in/shift left/serial out



(c) Parallel in/serial out



(d) Serial in/parallel out



(e) Parallel in/parallel out



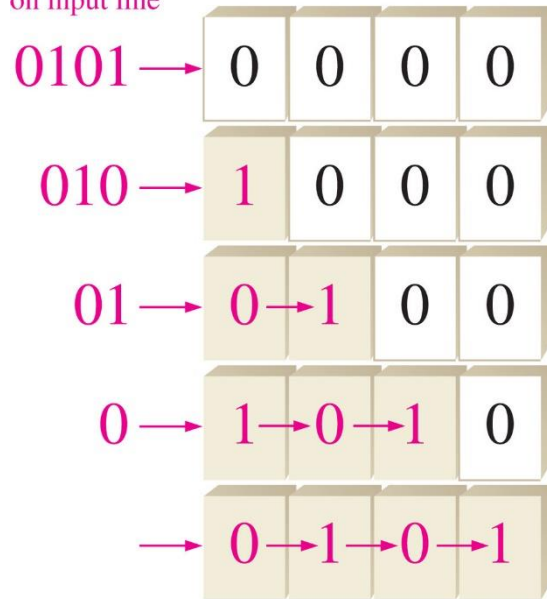
(f) Rotate right



(g) Rotate left

Shift Register Types – From Last Week

Serial bits
on input line



Initially, the register contains only *invalid* data or all zeros as shown here.

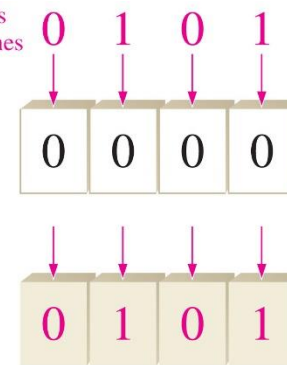
First bit (1) is shifted serially into the register.

Second bit (0) is shifted serially into register and first bit is shifted right.

Third bit (1) is shifted into register and the first and second bits are shifted right.

Fourth bit (0) is shifted into register and the first, second, and third bits are shifted right. The register now stores all four bits and is full.

Parallel bits
on input lines



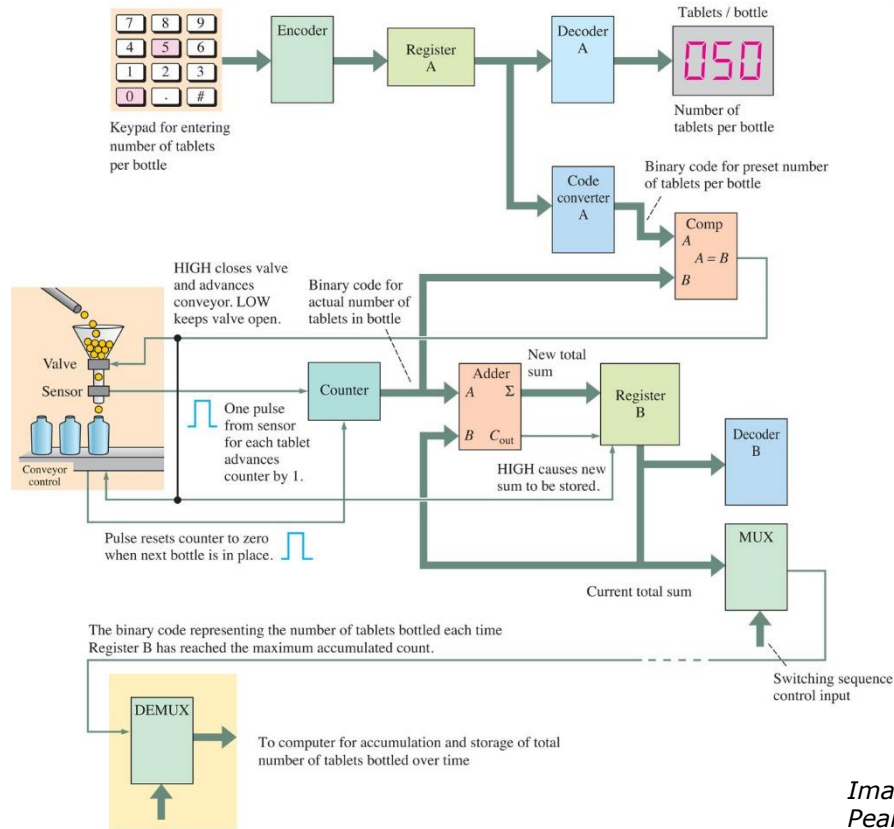
Initially, the register is empty, containing only nondata zeros.

All bits are shifted in and stored simultaneously.

Counters

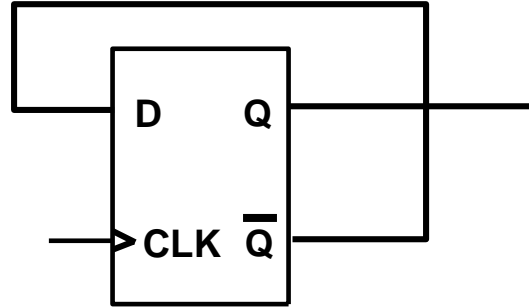
- A **counter** is a multi-bit register that goes through a determined sequence of states (values) upon the application of input (clock) pulses
- A counter which follows binary number sequence is a **binary counter**
 - **n-bit binary counter**
 - Has n flip-flops
 - Can count from 0 to $2^n - 1$ thus has 2^n different states
 - e.g. 2-bit binary counter will count
00 \rightarrow 01 \rightarrow 10 \rightarrow 11 \rightarrow 00 \rightarrow ...
(i.e. 0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 0 \rightarrow ...)

Digital System Application



One bit counter

To be completed in class

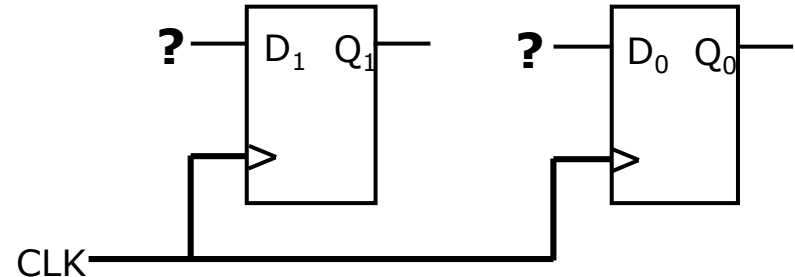


4-bit Binary Counter – Counting Sequence

| Bit 3 | Bit 2 | Bit 1 | Bit 0 | Value |
|-------|-------|-------|-------|-------|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 2 |
| 0 | 0 | 1 | 1 | 3 |
| 0 | 1 | 0 | 0 | 4 |
| 0 | 1 | 0 | 1 | 5 |
| 0 | 1 | 1 | 0 | 6 |
| 0 | 1 | 1 | 1 | 7 |
| 1 | 0 | 0 | 0 | 8 |
| 1 | 0 | 0 | 1 | 9 |
| 1 | 0 | 1 | 0 | 10 |
| 1 | 0 | 1 | 1 | 11 |
| 1 | 1 | 0 | 0 | 12 |
| 1 | 1 | 0 | 1 | 13 |
| 1 | 1 | 1 | 0 | 14 |
| 1 | 1 | 1 | 1 | 15 |
| 0 | 0 | 0 | 0 | 0 |

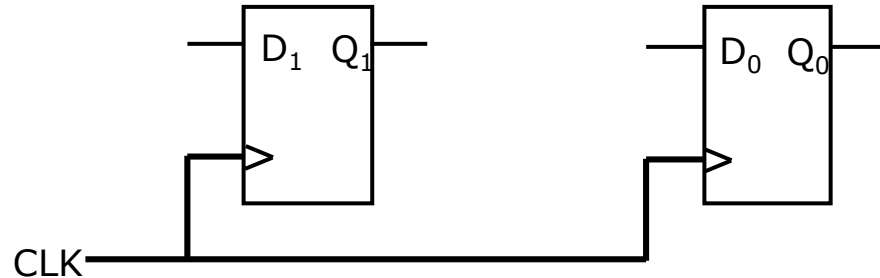
State

- Values stored in the flip-flops can be considered the **current state** of the circuit
- D inputs to the flip-flops are the **next state**
- D inputs are some function of the current state and inputs



Counter Example

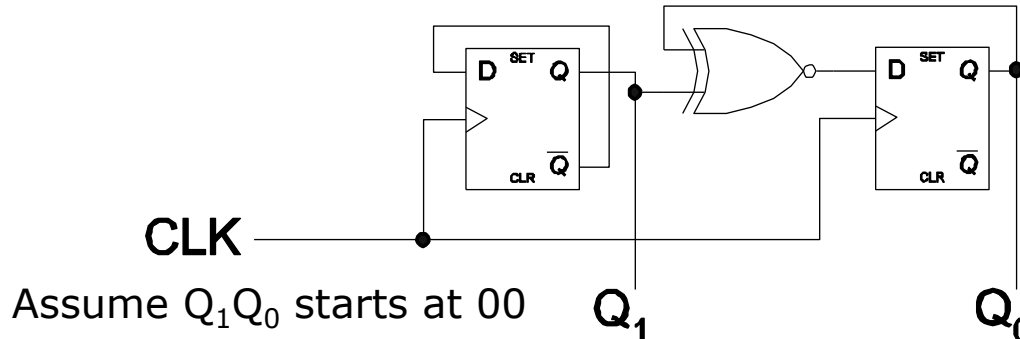
| Current State | | Next State | |
|---------------|-------|------------|-------|
| Q_1 | Q_0 | D_1 | D_0 |
| 0 | 0 | | |
| 0 | 1 | | |
| 1 | 0 | | |
| 1 | 1 | | |



To be completed in class

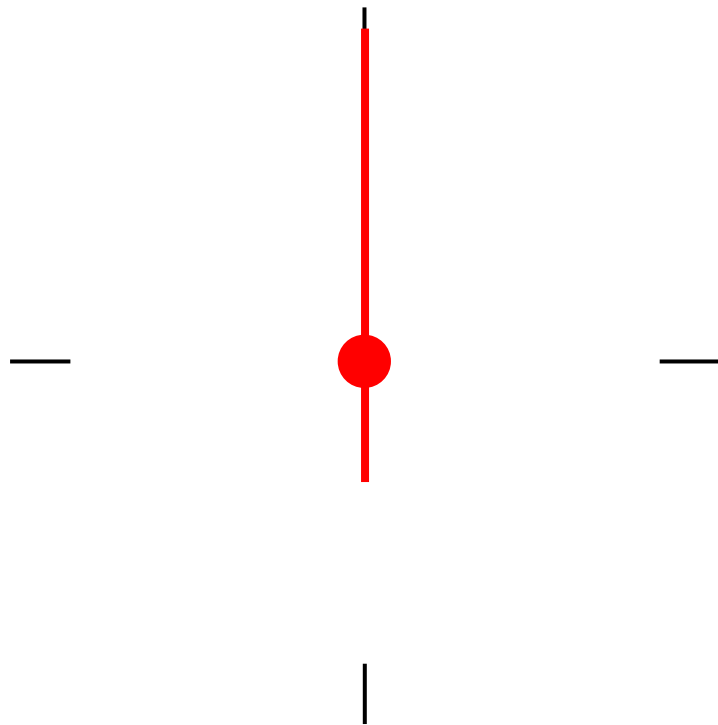
What sequence does the counter below count through?

- 0% **A.** $Q_1 Q_0$: $00 \rightarrow 11 \rightarrow 10 \rightarrow 01 \rightarrow 00 \rightarrow \dots$
- 0% **B.** $Q_1 Q_0$: $00 \rightarrow 10 \rightarrow 11 \rightarrow 01 \rightarrow 00 \rightarrow \dots$
- 0% **C.** $Q_1 Q_0$: $00 \rightarrow 11 \rightarrow 01 \rightarrow 10 \rightarrow 00 \rightarrow \dots$
- 0% **D.** $Q_1 Q_0$: $00 \rightarrow 01 \rightarrow 11 \rightarrow 10 \rightarrow 00 \rightarrow \dots$
- 0% **E.** $Q_1 Q_0$: $00 \rightarrow 01 \rightarrow 10 \rightarrow 11 \rightarrow 00 \rightarrow \dots$
- 0% **F.** $Q_1 Q_0$: $00 \rightarrow 10 \rightarrow 01 \rightarrow 11 \rightarrow 00 \rightarrow \dots$



Short Break

- Stand up and stretch



Counters

- Key points
 - Next state is a function of previous state (and possibly inputs)
 - Count sequence can be binary numbers but does not have to be
 - If it is, counter is a **binary counter**
 - Circuits are **synchronous**
 - All flip-flops have the same clock
 - There are asynchronous counters as well – we won't discuss these

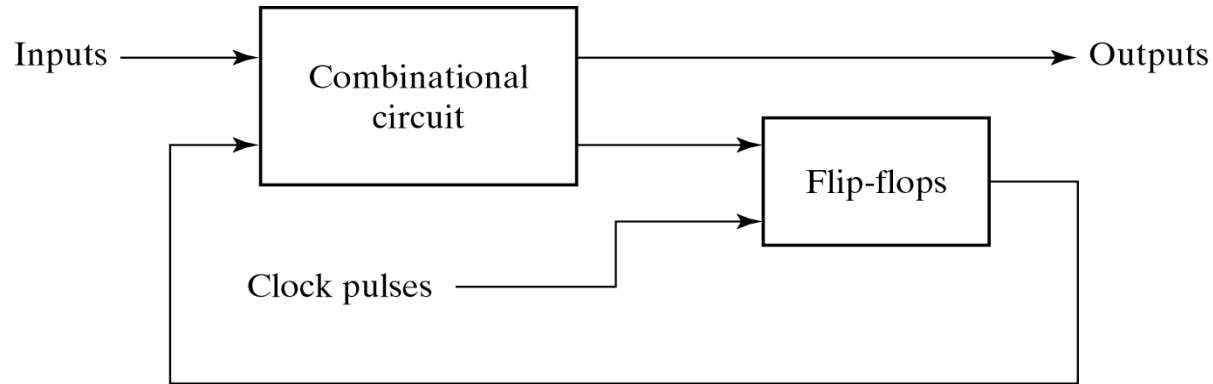
Example

(to be worked through in class)

- 2-bit counter that counts $00 \rightarrow 10 \rightarrow 01 \rightarrow 00 \rightarrow$

Next Time

- More sequential circuits
 - State Machines



Note that a counter may or may not have external inputs.