Lecture 12: State Machines 2

Digital Electronics 1

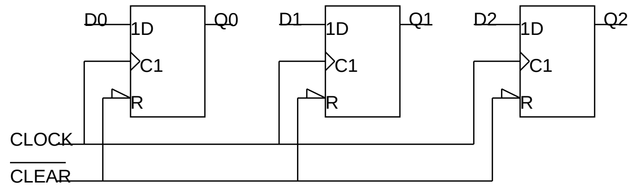
Points Addressed in this Lecture

- Registers and shift registers
- Synchronous and asynchronous counters

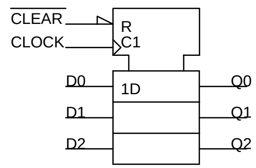
Registers

- A register is a digital electronic device capable of storing several bits of data
  - Normally made from D-type flip-flops with asynchronous RESET inputs
  - Operates on the bits of the data word in parallel (parallel in / parallel out)
- Operation
  - Data on each data input is stored in the flip-flop on the rising edge of CLOCK
  - The data can be read from the Q outputs
  - New data can be reloaded by re-CLOCKing the register
  - The register can be cleared (zeroed) by asserting the CLEAR inputs

3-bit Parallel in/Parallel out

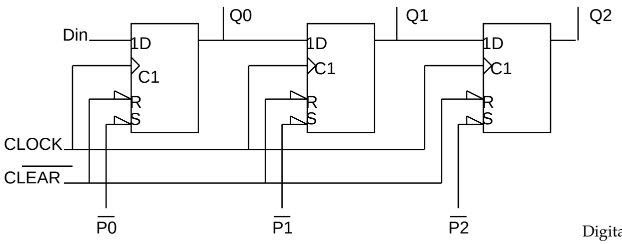


- Symbol



Shift Registers

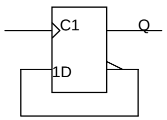
- Application Examples
  - Multiplication and division by integer power of 2
  - Conversion of data between parallel formats and bit-serial formats
- Construction
  - Like ordinary registers but with Q outputs connected to D inputs of the following flip-flop
  - E.g.: 3-bit shift register



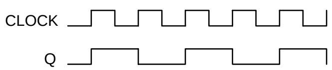
- Operation
  - On the rising edge of the clock, each bit moves right by one flip-flop
  - All the flip-flops can be asynchronously reset
  - Parallel data can be asynchronously loaded into flip-flops using the P signals
  - The data at the output of each flip-flop can be read from the Q signals hence
    - bits can be input serially at Din and output serially from Q2 with a delay of 3 clocks
    - bits can be input serially at Din an output in parallel from Q2:0 after 3 clocks

Divide by 2 Circuit

- Consider a D-type flip-flop with $\overline{Q}$ connected to D



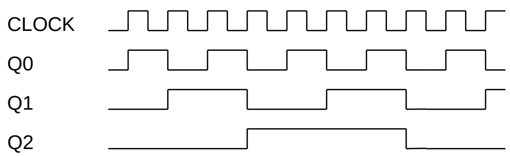
- D is always the inverse of Q hence Q will always toggle on a rising clock edge



- The frequency of Q is half the frequency of CLOCK

Binary Counters

- Example: 3-bit counter

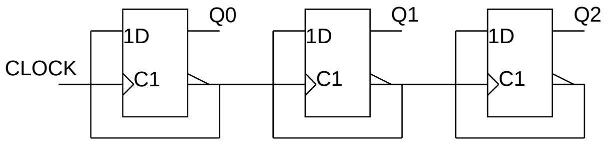


Decimal	Q2	Q1	Q0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

- Note that (in frequency terms)
  - Q0 = CLOCK / 2
  - Q1 = Q0 / 2
  - Q2 = Q1 / 2

Asynchronous Binary Counter

- Asynchronous counters are made from flip-flops
  - NOT all clocked with the same clock
  - sometimes called ripple counters
  - can be implemented using divide by 2 circuits
  - e.g. 3 bit counter



- Notes
  - called asynchronous because the C1 inputs of the flip-flops are not all driven by the same (CLOCK) signal
  - each output depends on a change in the previous flip-flops output
  - sometimes called a ripple counter because the data "ripples" from the output of one flip-flop to the input of the next
  - can also be implemented in JK

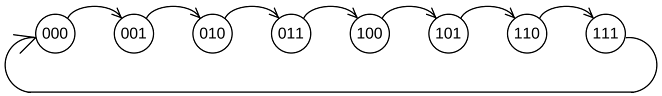
- Limitations
  - Consider the change from count 3 to count 4
    1. CLOCK goes from low to high
    2. Q0 goes from high to low
    3. Q1 goes from high to low
    4. Q2 goes from low to high
- The "CLOCK-TO-Q" delay of a typical flip-flop is about 30 ns
  - Hence total time needed is about 90 ns.
  - Hence max CLOCK frequency is = 11.1 MHz
  - The time needed for such transitions increases with the number of bits in the counter

Synchronous Binary Counters

- All flip-flops clocked with the same signal
  - hence all outputs change simultaneously
- the sequence of the count is controlled by combinational logic
  - sometimes called the state sequence
  - note that synchronous binary counters use both sequential and combinational elements

Example: 3-bit up-counter

- State Diagram

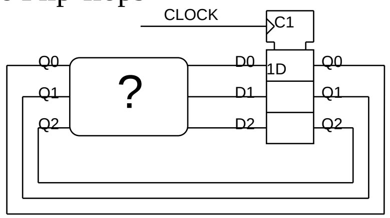


- From the state diagram we can construct a state transition table (let the variables be A (MSB), B and C)

Current State			Next State		
A	B	C	A+	B+	C+
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0
1	1	0	1	1	1
1	1	1	0	0	0

Implementation

- Use a register of flip-flops + Logical combinations of state variables
- E.g. For D-type Flip-flops



- The design task is to find a combinational circuit for ?
  - It should give the D inputs from the Q outputs such that, on the next clock, the correct counting sequence is followed
  - Other flip-flops can be used, e.g. J-K.