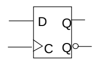
Registers and Counters

Last Lecture

- Last time we looked at:
- latches
- flip flops
- We saw that these devices hold a value depending on their inputs.
- A data input value is loaded into the register on the rise of the clock edge.
- Some circuits have additional ~clear or ~reset inputs.



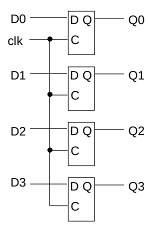
Positive edge-triggered flip-flop

C	D	Q
↑	0	0
↑	1	1
↓	?	Qn

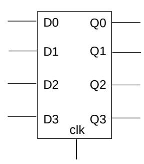
Registers from Latches

- Often want to read a group of data inputs into a set of latches at the same time (e.g. reading a 4 bit value off a computer bus)
- A group of latches can be combined to form a register
- A 4 bit register can be made from 4 latches

Four bit Register



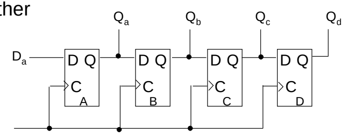
Circuit



Symbol

Shift Registers

- Allow stored data to be moved from one bit position to another



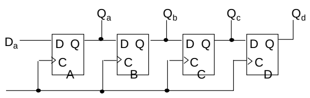
		Q_a	Q_b	Q_c	Q_d
D_a =1:	Initial Values	0	0	1	0
	after clock 1	1	0	0	1
	after clock 2	1	1	0	0
D_a = 0:	after clock 3	0	1	1	0
	after clock 4	0	0	1	1
	after clock 5	0	0	0	1

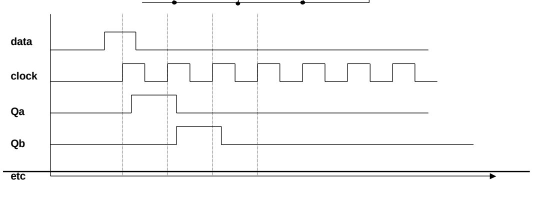
Shift Registers

- Points to note:
  - At every clock pulse, the first flip flop is loaded with the value of the data in stream
  - The data that was in this flip flop is then loaded into the second and so on.
  - The data can be taken out of the last flip flop in serial form or it can be taken from all outputs at the same time – parallel form.

Shift Registers

- For each flip flop there is a delay between clock pulse and output. This delay provides time for the next flip flop in the chain to load the data from the previous stage.





Shift Registers

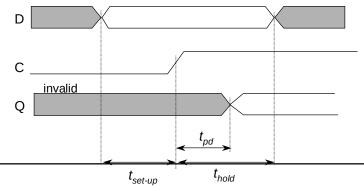
- Shift registers can also be loaded using parallel input lines
- Therefore inputs can be parallel or serial
- Outputs can be parallel or serial
- Functions that shift registers can carry out include:
  - Serial Loading
  - Serial Output
  - Parallel Output
  - Parallel Loading
- This makes them suitable for a wide variety of tasks

Shift Registers

- A large variety of integrated circuit (IC) shift registers are available with various combinations of
  - serial and/or parallel input
  - serial and/or parallel output
  - shift left and/or shift right
- Applications include:
  - Converting a parallel word into serial form or vise versa
  - performing a number of logical and arithmetic operations (binary multiplication/division involves shifting).

Registers

- Timing characteristics for edge-triggered registers
  - propagation delay (t_{pd}) defined as the time between the clock edge and the output changing
  - Set-up (t_{setup}) and hold (t_{hold}) times are the times which the data must be held steady before and after the clock edge



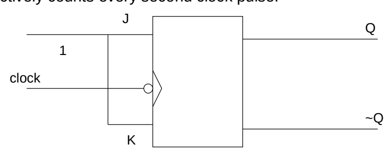
Counters

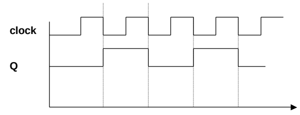
- Counters may be clocked by:
  - regular clock pulses to determine a certain time duration
  - random pulses to count the occurrences of a particular event
- Most common counters count natural binary sequence

up counter	down counter
000	111
001	110
010	101
011	100
100	011
101	010
110	001
111	000
000	111
...	...

JK Flip flop - recall

- A JK flip flop toggles when both inputs are 1. In this case it effectively counts every second clock pulse:



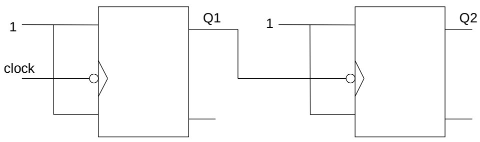


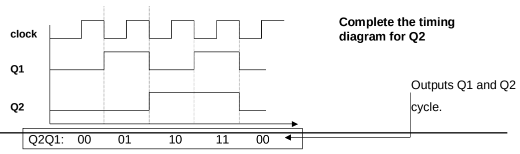
You can also say it counts from 0 to 1 and back again.

Sometimes called a "scale of 2 counter"

Counters

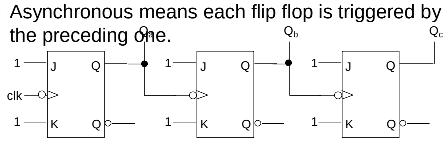
- Connect two such flip flops together:



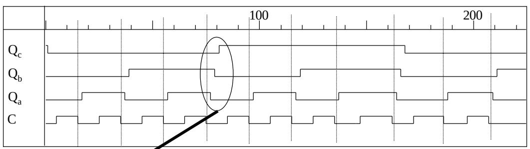


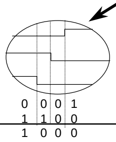
Counters

- Ripple (asynchronous) Counter
  - e.g. A 3 bit ripple counter using negative edge triggered JK flip-flops
  - Asynchronous means each flip flop is triggered by the preceding one.



Counters

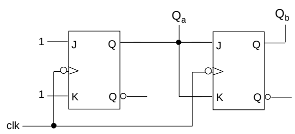


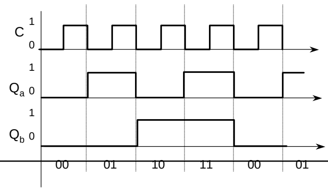


- These propagation delays cause a number of sequence changes when going from one number to the next, which can be undesirable in a lot of situations.
- The counter can get the wrong value at a particular instant in time
- Operating speed is therefore limited.

Counters

- Synchronous counters
  - Outputs of all the flip-flops change at the same time
  - e.g. a 2-bit synchronous counter

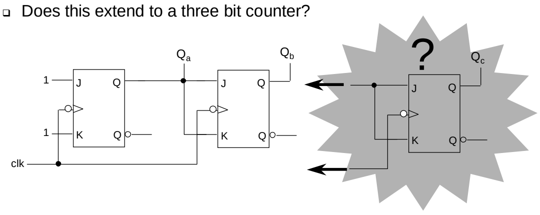




Page 4

Counters

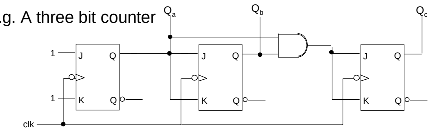
- Synchronous counters
  - Does this extend to a three bit counter?



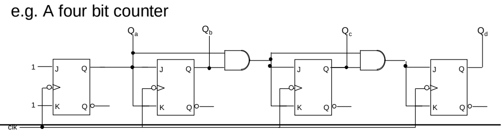
  - Q_c should not toggle until both Q_a and Q_b are 1

Counters

- Synchronous counters
  - e.g. A three bit counter

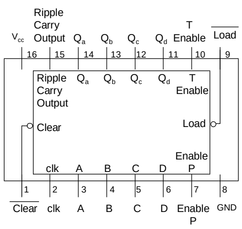


  - e.g. A four bit counter



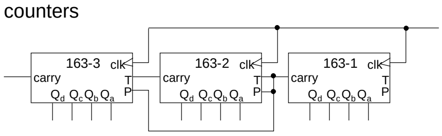
Counters

- An integrated circuit counter
  - Many forms of counters available as integrated circuits
  - e.g. 74163 - four bit synchronous binary up counter
  - **Clock:** count advanced by 1 on ↑
  - **$\overline{\text{Clear}}$:** when = 0 count reset to 0 on next clk
  - $\overline{\text{Load}}$: when = 0 count set to values on A-D on next clk
  - **Enable T & P:** Counting disabled when either is equal 0
  - **Q_a, Q_b, Q_c, Q_d:** state of the counter
  - **Ripple Carry output :** = 1 when count = 1111 otherwise 0



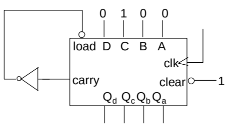
Counters

- An integrated circuit counter
  - In many cases more than 16 states required for counting
  - counters like 163 can be cascaded to form larger counters

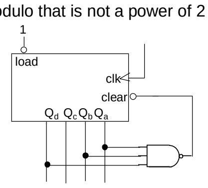


Counters

- Modulo-n counters
  - A **modulo-n** counter generates n states before it repeats itself
  - e.g. a 2 bit count is modulo 4 and a 4 bit counter is modulo 16
  - Often a counter which has a modulo that is not a power of 2 is required e.g. modulo 12



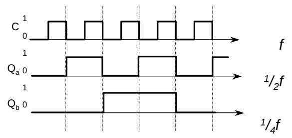
Counts from 4 through to 15 before being set to 4 again



Counts from 0 through to 11 before being reset

Counters

- Frequency division
  - Binary counters offer the possibility of frequency division



  - Often used where clocks of different frequencies are required in a circuit or to produce a compact accurate low frequency oscillator (e.g. an oscillator for a digital watch)

Registers/Shift registers and Counters

- Summary
  - Registers consist of a group of D-type latches or flip-flops which are clocked simultaneously to store a binary word
    - set-up and hold times must be observed
  - Shift registers allow data to be moved from one bit position to another
    - used for parallel↔serial conversion and some types of arthmetic operations
  - Counting is a common requirement in sequential logic circuits
    - Counters can be asynchronous or synchronous
    - Many IC packages exist which implement counters

What you should be able to do

- Explain the operation of a register
- Explain the propagation delays associated with registers
- Outline the use of registers for converting serial/parallel inputs/outputs.
- Explain the operation of ripple (asynchronous counters)
- Explain the operation of synchronous counters
- Outline the characteristics of modulo n counters.