Counters

CASCADING & FREQUENCY DIVISION

Cascading Counters

* Counters are cascaded to create larger counters
* Used to create higher MOD counters
* MOD will be used to determine frequency division

Cascaded Counters and MOD

* The effective MOD of the cascaded counter is the product of all the MODs of each counter
  - A MOD 4 counter cascaded with a MOD 8 counter creates a MOD 32 counter
  - MOD is the total number of counts a counter can count to

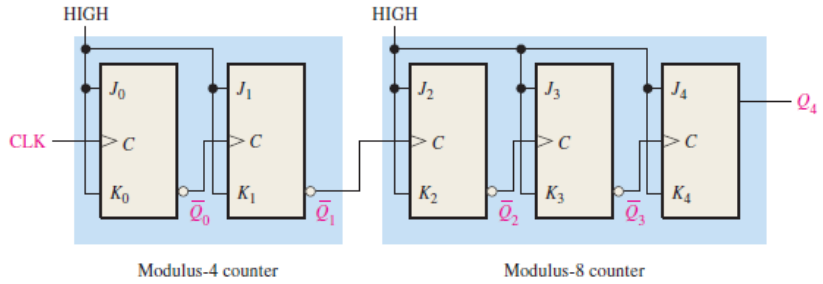
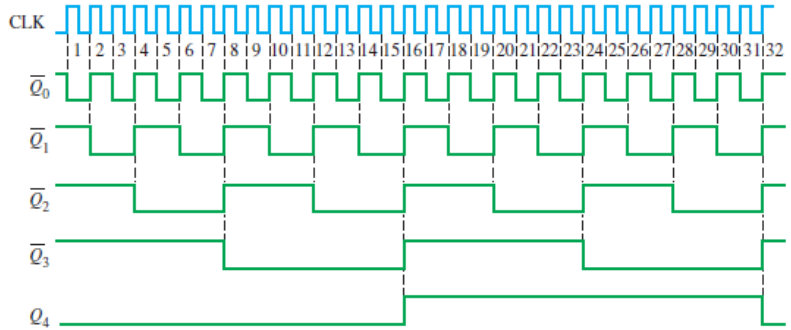
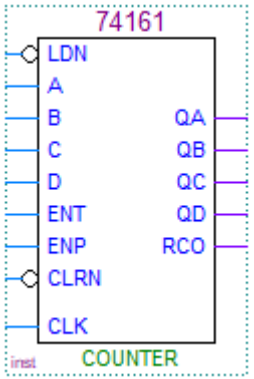


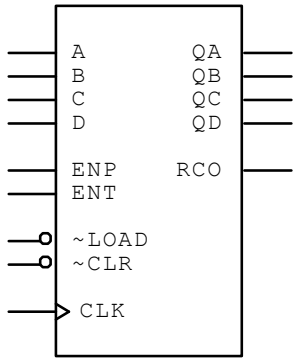
FIGURE 9–35 Two cascaded asynchronous counters (all *J* and *K* inputs are HIGH).



4 – Bit Synchronous Counter

* The 4 – bit synchronous counter in Quartus is representative of a counter IC
* Has additional inputs and output
  - ENP, ENT, Load (LDN), Clear (CLRN), A, B, C, D
  - RCO

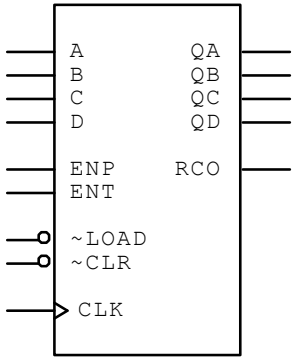




Alternative Schematic Symbol

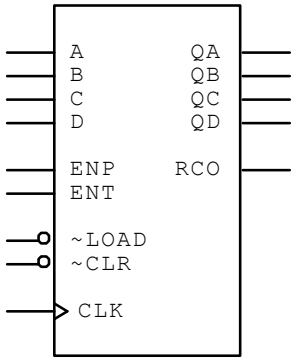
Inputs

* A, B, C, D are initial values to load into the counter (1 for each flip flop)
  - Use if the counter starts at a specific value, A is LSB
* Load will place the initial value set by A, B, C, and D inputs to the counter on next clock edge
* ENP and ENT are enables, both must be HIGH for counter to work
* Clear will reset the counter to 0000



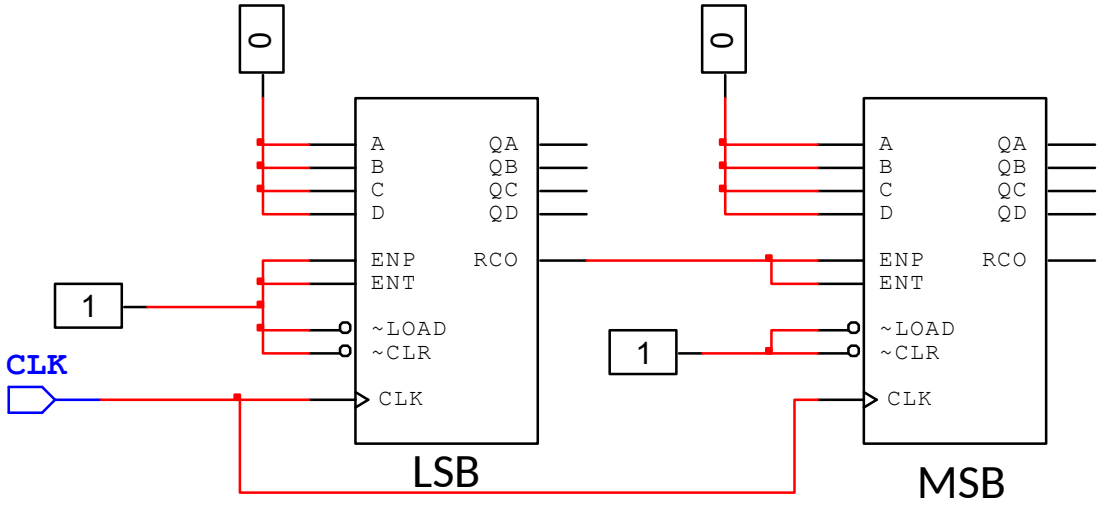
Outputs

* QA, QB, QC, and QD are the output count with QA being the LSB
* ROC is the rollover output that goes high when the counter reaches it’s terminal count
  - Output will go LOW on next clock pulse after terminal count is reached



Cascading 4-Bit Synchronous Counters

* Need to use the additional inputs and outputs from synchronous counter
  - LSB counter is always active
  - The MSB counter is active only when a terminal count is reached on the LSB counter



8 – bit synchronous counter

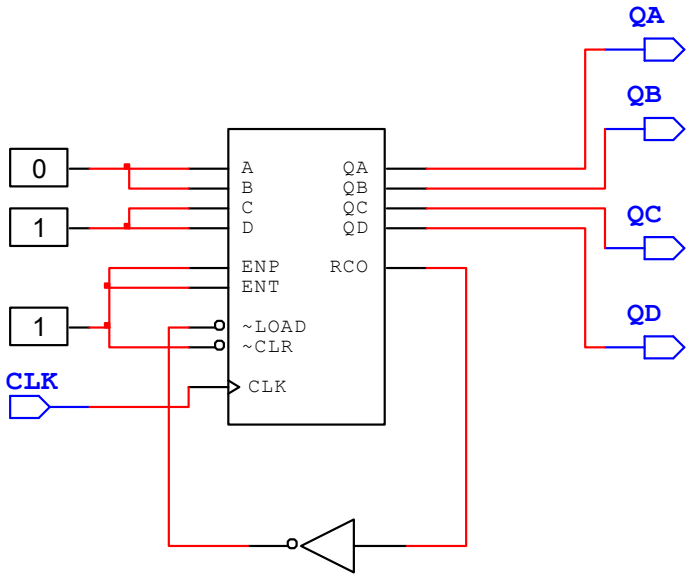
Truncated MOD

* What if you don’t want to use the full MOD of a synchronous counter?
  (and don’t want to create a custom one with state diagrams)
* Use the load and initial count inputs to specify the MOD of a counter

Truncated MOD

* A single 4 – bit counter is MOD 16, how can it be set up to create a MOD 4?
  - What is the difference between max MOD and desired MOD?
  - The difference is the input to the preset inputs
  - Use the RCO output to reload the initial count value
* How can a MOD 4 counter be created from a synchronous 4 – bit counter?

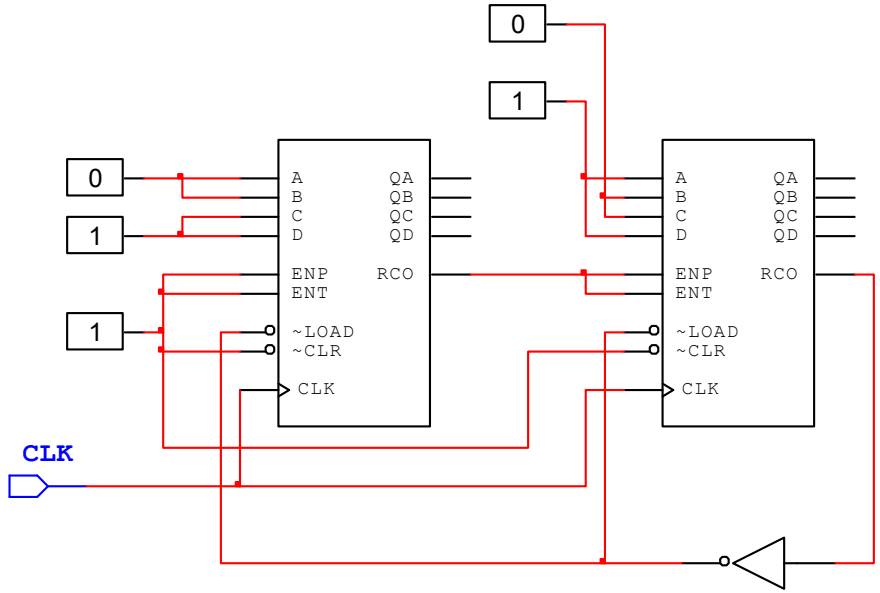
MOD 4 Counter



Example

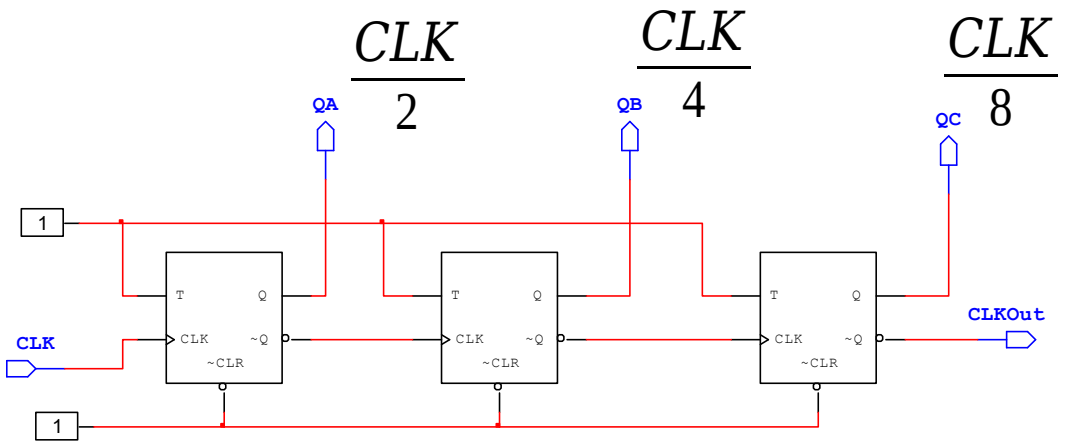
* How would a synchronous 100 MOD counter be created?

MOD 100


0
1
0
1
1
A
B
C
D
QA
QB
QC
QD
ENP
ENT
RCO
~LOAD
~CLR
CLK
A
B
C
D
QA
QB
QC
QD
ENP
ENT
RCO
~LOAD
~CLR
CLK
CLK

Frequency Divider

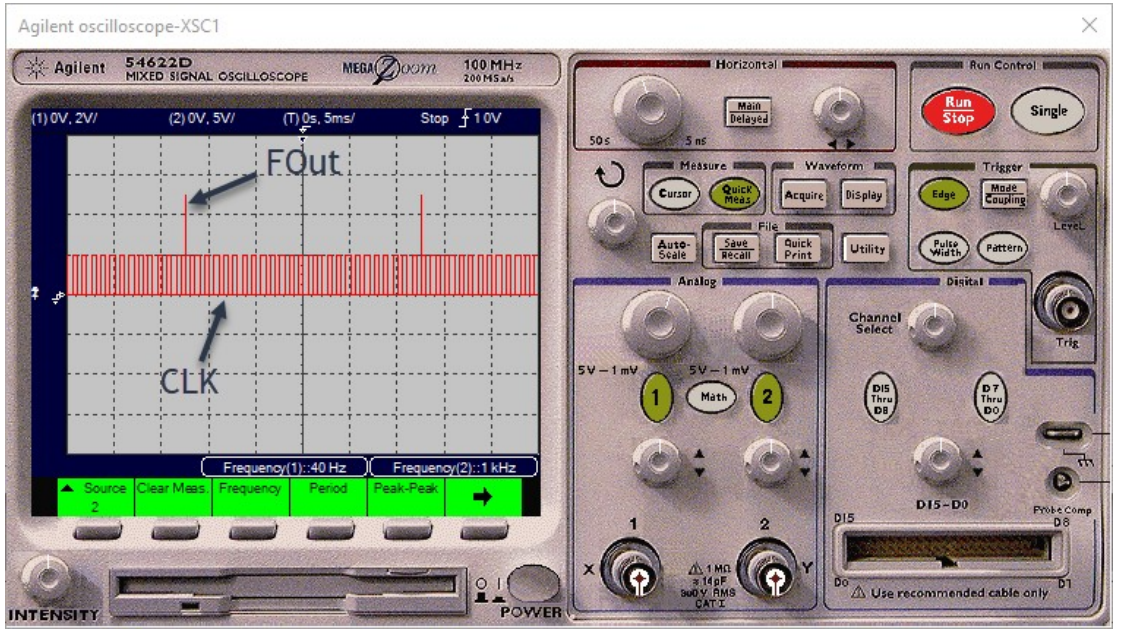
* Any counter will divide the input clock on the output of the flip flops
* Can use the output of counter as another clock elsewhere in digital system



If CLK is 80 Hz, what is the possible clock frequencies that can be used?

MOD and Frequency Divider

* The MOD value is the value a counter can divide the frequency of a clock
* For a 1KHz clock put into a 25 MOD counter, the counter will divide the clock by 25
  - For every 25 pulses on the clock, 1 pulse is generated
  - 40 Hz on signal on output



Example

* Create a frequency divider to divide a clock by 20 using T flip flops.

Example

* Create a frequency divider to divide a clock by 50 using 4-bit sync. counters

Potential Problems

*The output of the RCO (sync.) or the NAND gate (async.) to reset the counter is typically very small in width

- Can create problems is using these for a clock signal on another circuit

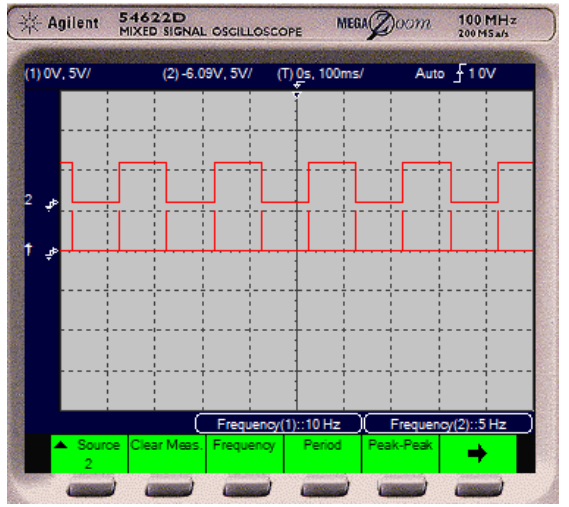
* Want a clock to have equal time high and equal time low

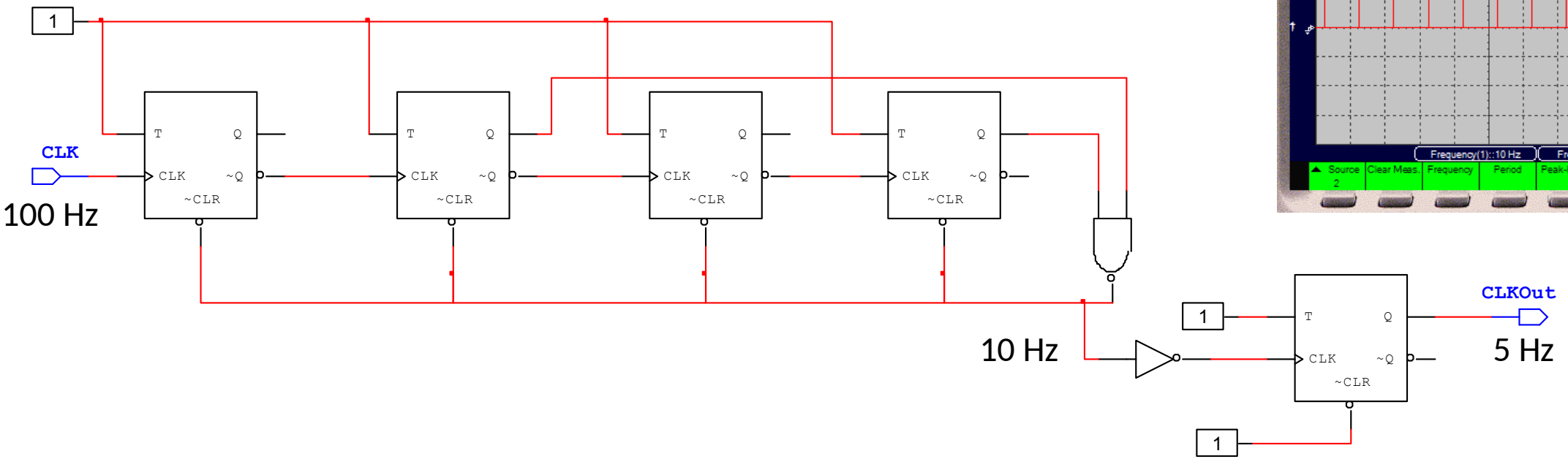
"Distorted" Clock

Ideal Clock

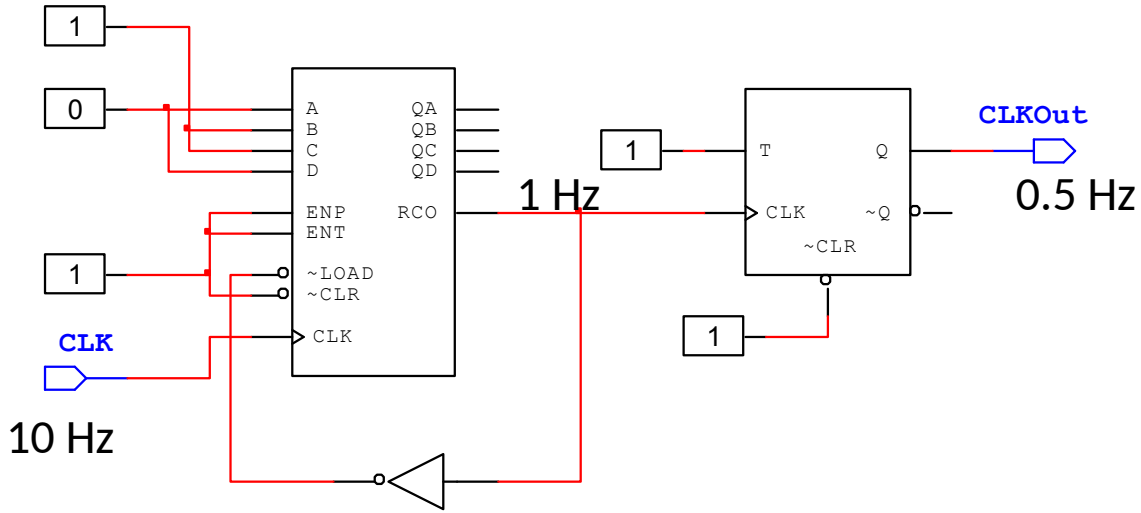
Creating Clock Signal

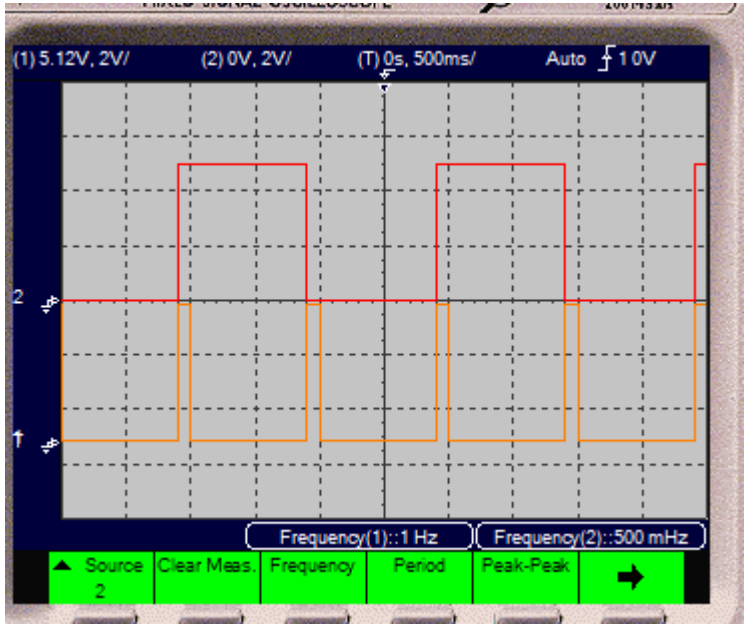
* Using a T flip flop will make the output equal HIGH and LOW time
  * Placing through a flip flop further divides frequency by 2





Sync. Example

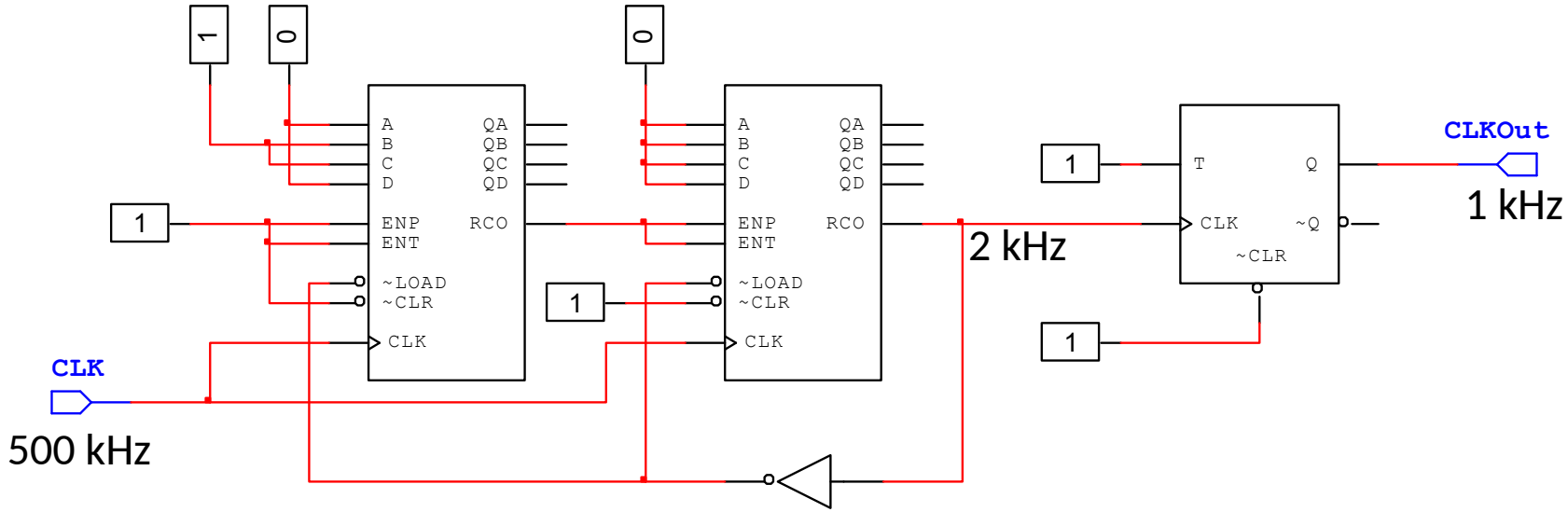




Example

* Use multiple synchronous 4 – bit counters and other logic elements to create a clock divider of 500. The output clock should have equal HIGH and LOW times.

Example


1
0
0
A
B
C
D
QA
QB
QC
QD
ENP
ENT
RCO
~LOAD
~CLR
CLK
1
1
2 kHz
T
Q
CLK
~Q
~CLR
1
CLKOut
1 kHz
CLK
500 kHz

Example

* A system has a 1 MHz master clock. A subsystem can only run at a max frequency of 30 kHz. Design the frequency divider for the subsystem clock using a 4-bit synchronous counter. The output should have equal HIGH and LOW time.