CMOS Oscillators

Fairchild Semiconductor
Application Note 118
October 1974

FAIRCHILD SEMICONDUCTOR TM

INTRODUCTION

This note describes several square wave oscillators that can be built using CMOS logic elements. These circuits offer the following advantages:

- Guaranteed startability
- Relatively good stability with respect to power supply variations
- Operation over a wide supply voltage range (3V to 15V)
- Operation over a wide frequency range from less than 1 Hz to about 15 MHz
- Low power consumption (see AN-90)
- Easy interface to other logic families and elements including TTL

Several RC oscillators and two crystal controlled oscillators are described. The stability of the RC oscillator will be sufficient for the bulk of applications; however, some applications will probably require the stability of a crystal. Some applications that require a lot of stability are:

1. Timekeeping over a long interval. A good deal of stability is required to duplicate the performance of an ordinary wrist watch (about 12 ppm). This is, of course, obtainable with a crystal. However, if the time interval is short and/or the resolution of the timekeeping device is relatively large, an RC oscillator may be adequate. For example: if a stopwatch is built with a resolution of tenths of seconds and the longest interval of interest is two minutes, then an accuracy of 1 part in 1200 (2 minutes x 60 seconds/minute x 10 tenth/second) may be acceptable since any error is less than the resolution of the device.
2. When logic elements are operated near their specified limits. It may be necessary to maintain clock frequency accuracy within very tight limits in order to avoid exceeding the limits of the logic family being used, or in which the timing relationships of clock signals in dynamic MOS memory or shift register systems must be preserved.
3. Baud rate generators for communications equipment.
4. Any system that must interface with other tightly specified systems. Particularly those that use a “handshake” technique in which Request or Acknowledge pulses must be of specific widths.

LOGICAL OSCILLATORS

Before describing any specific circuits, a few words about logical oscillators may clear up some recurring confusion.

Any odd number of inverting logic gates will oscillate if they are tied together in a ring as shown in *Figure 1*. Many beginning logic designers have discovered this (to their chagrin) by inadvertently providing such a path in their designs. However, some people are confused by the circuit in *Figure 1* because they are accustomed to seeing sinewave oscillators implemented with positive feedback, or amplifiers with non-inverting gain. Since the concept of phase shift becomes a little strained when the inverters remain in their linear region for such a short period, it is far more straightforward to analyze the circuit from the standpoint of ideal switches with finite propagation delays rather than as amplifiers with 180° phase shift. It then becomes obvious that a “1” chases itself around the ring and the network oscillates.

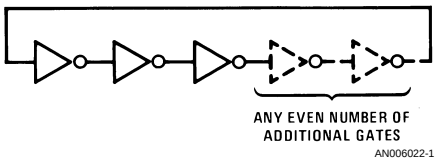


FIGURE 1. Odd Number of Inverters Will Always Oscillate

The frequency of oscillation will be determined by the total propagation delay through the ring and is given by the following equation.

$$f = \frac{1}{2nTp}$$

Where:

f = frequency of oscillation

Tp = Propagation delay per gate

n = number of gates

This is not a practical oscillator, of course, but it does illustrate the maximum frequency at which such an oscillator will run. All that must be done to make this a useful oscillator is to slow it down to the desired frequency. Methods of doing this are described later.

To determine the frequency of oscillation, it is necessary to examine the propagation delay of the inverters. CMOS propagation delay depends on supply voltage and load capacitance. Several curves for propagation delay for Fairchild’s 74C line of CMOS gates are reproduced in *Figure 3*. From these, the natural frequency of oscillation of an odd number of gates can be determined.

An example may be instructive.

Assume the supply voltage is 10V. Since only one input is driven by each inverter, the load capacitance on each inverter is at most about 8 pF. Examine the curve in *Figure 3c* that is drawn for V_{CC} = 10V and extrapolate it down to 8 pF. We see that the curve predicts a propagation delay of about 17 ns. We can then calculate the frequency of oscillation for three inverters using the expression mentioned above. Thus:

$$f = \frac{1}{2 \times 3 \times 17 \times 10^{-9}} = 9.8\ \text{MHz}$$

Lab work indicates this is low and that something closer to 16 MHz can be expected. This reflects the conservative nature of the curves in *Figure 3*.

Since this frequency is directly controlled by propagation delays, it will vary a great deal with temperature, supply voltage, and any external loading, as indicated by the graphs in *Figure 3*. In order to build a usefully stable oscillator it is nec-

essary to add passive elements that determine oscillation frequency and minimize the effect of CMOS characteristics.

STABLE RC OSCILLATOR

Figure 2 illustrates a useful oscillator made with three inverters. Actually, any inverting CMOS gate or combination of gates could be used. This means left over portions of gate packages can be often used. The duty cycle will be close to 50% and will oscillate at a frequency that is given by the following expression.

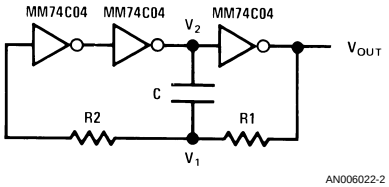


FIGURE 2. Three Gate Oscillator

$$f \cong \frac{1}{2\,R1\,C\left(\frac{0.405\,R2}{R1 + R2} + 0.693\right)}$$

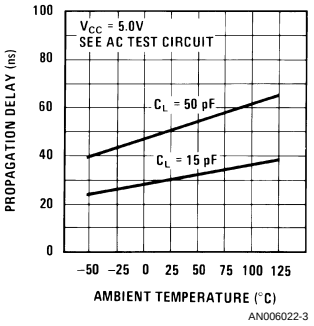
Another form if this expression is:

$$f \cong \frac{1}{2C\,(0.405\,R_{eq} + 0.693\,R1)}$$

Where:

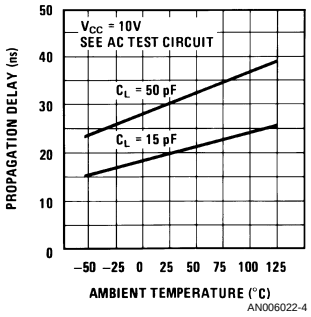
$$R_{eq} = \frac{R1\,R2}{R1 + R2}$$

Propagation Delay vs Ambient Temperature MM54C00/MM74C00, MM54C02/MM74C02, MM54C04/MM74C04



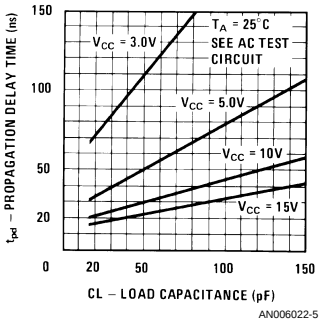
(a)

Propagation Delay vs Ambient Temperature MM54C00/MM74C00, MM54C02/MM74C02, MM54C04/MM74C04



(b)

Propagation Delay Time vs Load Capacitance MM54C00/MM74C00, MM54C02/MM74C02, MM54C04/MM74C04



(c)

FIGURE 3. Propagation Delay for 74C Gates

The following three special cases may be useful.

$$\text{If } R1 = R2 = R \qquad f \cong \frac{0.559}{RC}$$

$$\text{If } R2 >>> R1 \qquad f \cong \frac{0.455}{RC}$$

$$\text{If } R2 <<< R1 \qquad f \cong \frac{0.722}{RC}$$

Figure 4 illustrates the approximate output waveform and the voltage V_1 at the charging node.

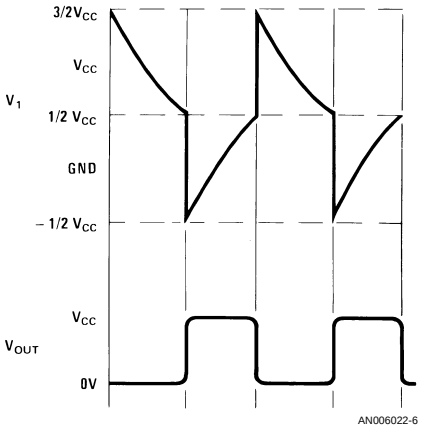


FIGURE 4. Waveforms for Oscillator in *Figure 2*

Note that the voltage V_2 will be clamped by input diodes when V_1 is greater than V_{CC} or more negative than ground. During this portion of the cycle current will flow through R2. At all other times the only current through R2 is a very minimal leakage term. Note also that as soon as V_1 passes through threshold (about 50% of supply) and the input to the last inverter begins to change, V_1will also change in a direction that reinforces the switching action; i.e., providing positive feedback. This further enhances the stability and predictability of the network.

This oscillator is fairly insensitive to power supply variations due largely to the threshold tracking close to 50% of the supply voltage. Just how stable it is will be determined by the frequency of oscillation; the lower the frequency the more stability and vice versa. This is because propagation delay and the effect of threshold shifts comprise a smaller portion of the overall period. Stability will also be enhanced if R1 is made large enough to swamp any variations in the CMOS output resistance.

TWO GATE OSCILLATOR WILL NOT NECESSARILY OSCILLATE

A popular oscillator is shown in *Figure 5a*. The only undesirable feature of this oscillator is that it may not oscillate. This is readily demonstrated by letting the value of C go to zero. The network then degenerates into *Figure 5b*, which obviously will not oscillate. This illustrates that there is some value of C1 that will not force the network to oscillate. The real difference between this two gate oscillator and the three gate oscillator is that the former must be forced to oscillate by the capacitor while the three gate network will always oscillate willingly and is simply slowed down by the capacitor. The three gate network will always oscillate, regardless of the value of C1 but the two gate oscillator will not oscillate when C1 is small.

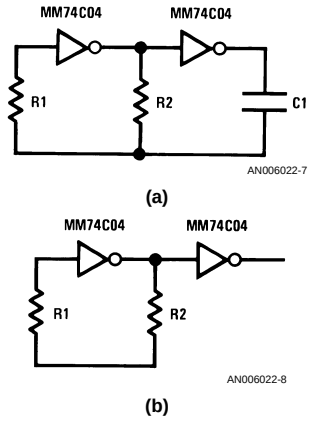


FIGURE 5. Less Than Perfect Oscillator

The only advantage the two gate oscillator has over the three gate oscillator is that it uses one less inverter. This may or may not be a real concern, depending on the gate count in each user's specific application. However, the next section offers a real minimum parts count oscillator.

A SINGLE SCHMITT TRIGGER MAKES AN OSCILLATOR

Figure 6 illustrates an oscillator made from a single Schmitt trigger. Since the MM74C14 is a hex Schmitt trigger, this oscillator consumes only one sixth of a package. The remaining 5 gates can be used either as ordinary inverters like the MM74C04 or their Schmitt trigger characteristics can be used to advantage in the normal manner. Assuming these five inverters can be used elsewhere in the system, *Figure 6* must represent the ultimate in low gate count oscillators.

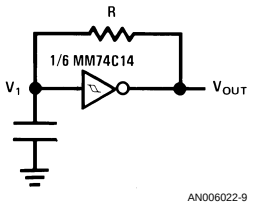


FIGURE 6. Schmitt Trigger Oscillator

Voltage V_1 is depicted in *Figure 7* and changes between the two thresholds of the Schmitt trigger. If these thresholds were constant percentages of V_{CC} over the supply voltage range, the oscillator would be insensitive to variations in V_{CC}. However, this is not the case. The thresholds of the Schmitt trigger vary enough to make the oscillator exhibit a good deal of sensitivity to V_{CC}.

Applications that do not require extreme stability or that have access to well regulated supplies should not be bothered by this sensitivity to V_{CC}. Variations in threshold can be expected to run as high as four or five percent when V_{CC} varies from 5V to 15V.

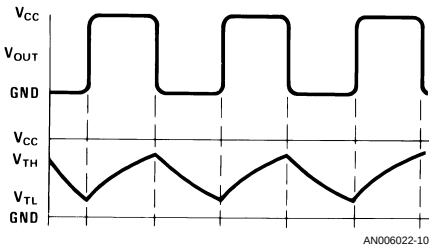


FIGURE 7. Waveforms for Schmitt Trigger Oscillator in *Figure 6*

A CMOS CRYSTAL OSCILLATOR

Figure 8 illustrates a crystal oscillator that uses only one CMOS inverter as the active element. Any odd number of inverters may be used, but the total propagation delay through the ring limits the highest frequency that can be obtained. Obviously, the fewer inverters that are used, the higher the maximum possible frequency.

CONCLUSIONS

A large number of oscillator applications can be implemented with the extremely simple, reliable, inexpensive and versatile CMOS oscillators described in this note. These oscillators consume very little power compared to most other approaches. Each of the oscillators requires less than one full package of CMOS inverters of the MM74C04 variety. Frequently such an oscillator can be built using leftover gates of the MM74C00, MM74C02, MM74C10 variety. Stability superior to that easily attainable with TTL oscillators is readily attained, particularly at lower frequencies. These oscillators are so versatile, easy to build, and inexpensive that they should find their way into many diverse designs.

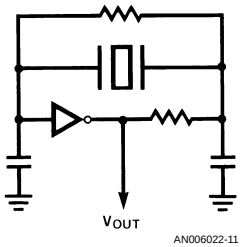


FIGURE 8. Crystal Oscillator

Fairchild Semiconductor Corporation
Americas
Customer Response Center
Tel: 1-888-522-5372

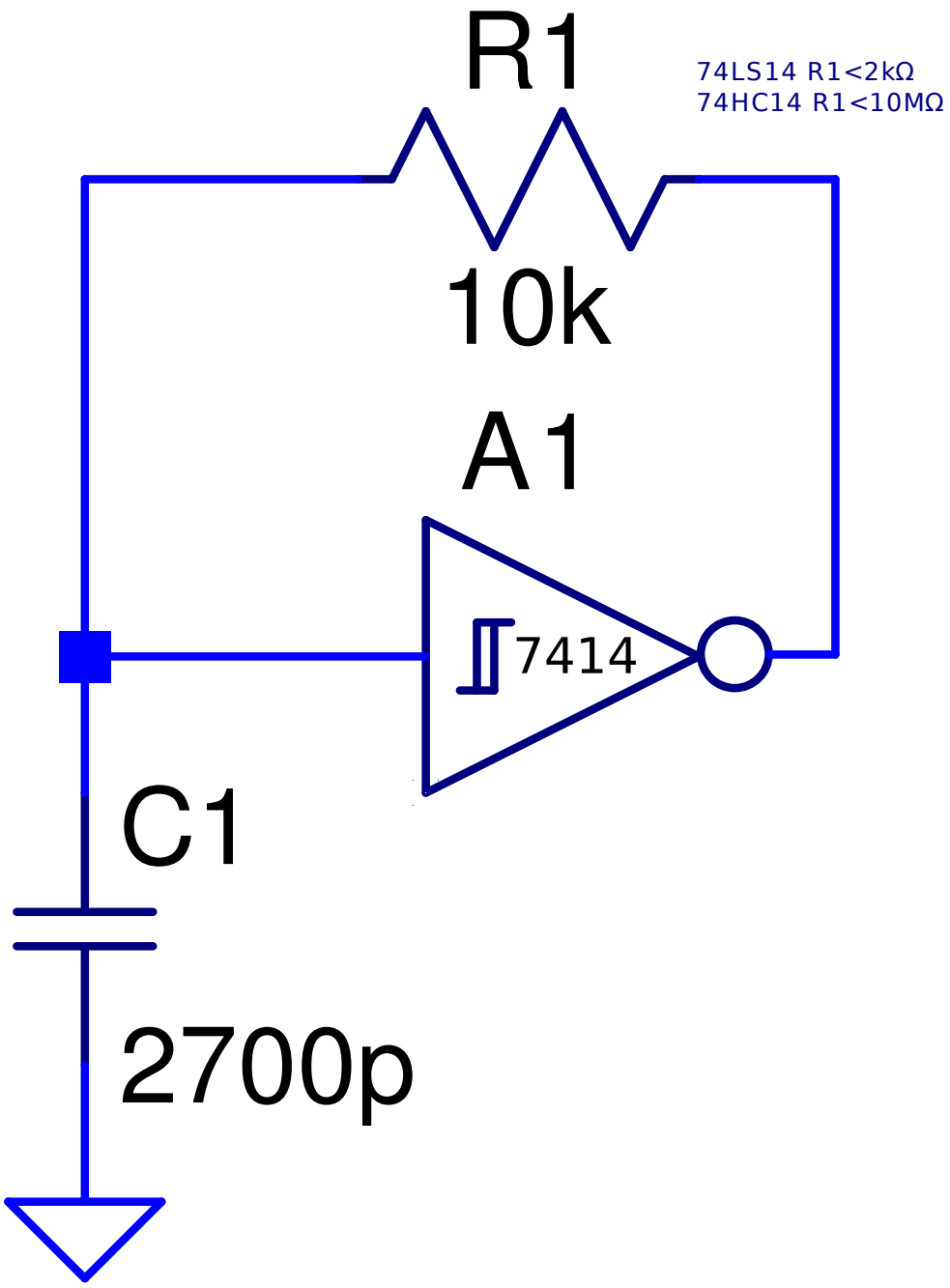
Fairchild Semiconductor Europe
Fax: +49 (0) 1 80-530 85 86
Email: europe.support@nsc.com
Deutsch Tel: +49 (0) 8 141-35-0
English Tel: +44 (0) 1 793-85-68-56
Italy Tel: +39 (0) 2 57 5631

Fairchild Semiconductor Hong Kong Ltd.
13th Floor, Straight Block,
Ocean Centre, 5 Canton Rd.
Tsimshatsui, Kowloon
Hong Kong
Tel: +852 2737-7200
Fax: +852 2314-0061

National Semiconductor Japan Ltd.
Tel: 81-3-5620-6175
Fax: 81-3-5620-6179

www.fairchildsemi.com

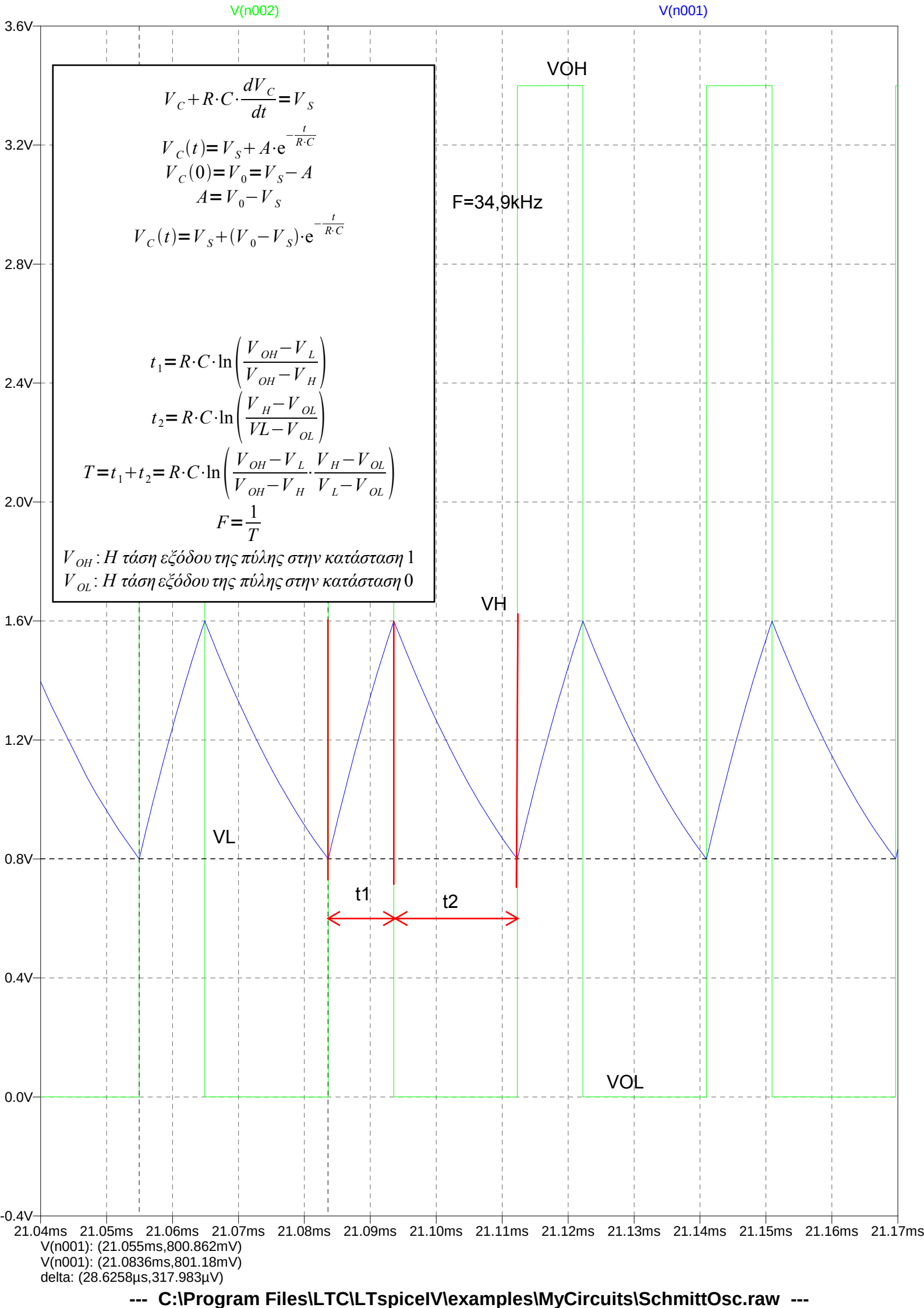
Schmit Trigger Oscillator



.tran 100ms uic

V_{OH}=3.4V V_L=0.8V V_H=1.6V

F=34.9kHz


V(n002)
V(n001)
3.6V
3.2V
2.8V
2.4V
2.0V
1.6V
1.2V
0.8V
0.4V
0.0V
-0.4V
$V_C + R \cdot C \cdot \frac{dV_C}{dt} = V_S$
$V_C(t) = V_S + A \cdot e^{-\frac{t}{R \cdot C}}$
$V_C(0) = V_0 = V_S - A$
$A = V_0 - V_S$
$V_C(t) = V_S + (V_0 - V_S) \cdot e^{-\frac{t}{R \cdot C}}$
$t_1 = R \cdot C \cdot \ln\left(\frac{V_{OH} - V_L}{V_{OH} - V_H}\right)$
$t_2 = R \cdot C \cdot \ln\left(\frac{V_H - V_{OL}}{VL - V_{OL}}\right)$
$T = t_1 + t_2 = R \cdot C \cdot \ln\left(\frac{V_{OH} - V_L}{V_{OH} - V_H} \cdot \frac{V_H - V_{OL}}{V_L - V_{OL}}\right)$
$F = \frac{1}{T}$
V_{OH} : Η τάση εξόδου της πύλης στην κατάσταση 1
V_{OL} : Η τάση εξόδου της πύλης στην κατάσταση 0
F=34,9kHz
VOH
VH
VL
VOL
t1
t2
21.04ms
21.05ms
21.06ms
21.07ms
21.08ms
21.09ms
21.10ms
21.11ms
21.12ms
21.13ms
21.14ms
21.15ms
21.16ms
21.17ms
V(n001): (21.055ms,800.862mV)
V(n001): (21.0836ms,801.18mV)
delta: (28.6258µs,317.983µV)
--- C:\Program Files\LTC\LTspiceIV\examples\MyCircuits\SchmittOsc.raw ---

A 4MHz Crystal Oscillator With An Inverter

LTSPICE Simulation Setup:
Control Panel->Hacks-> disable "Supply a min. inductor damping if no Rpar is given"

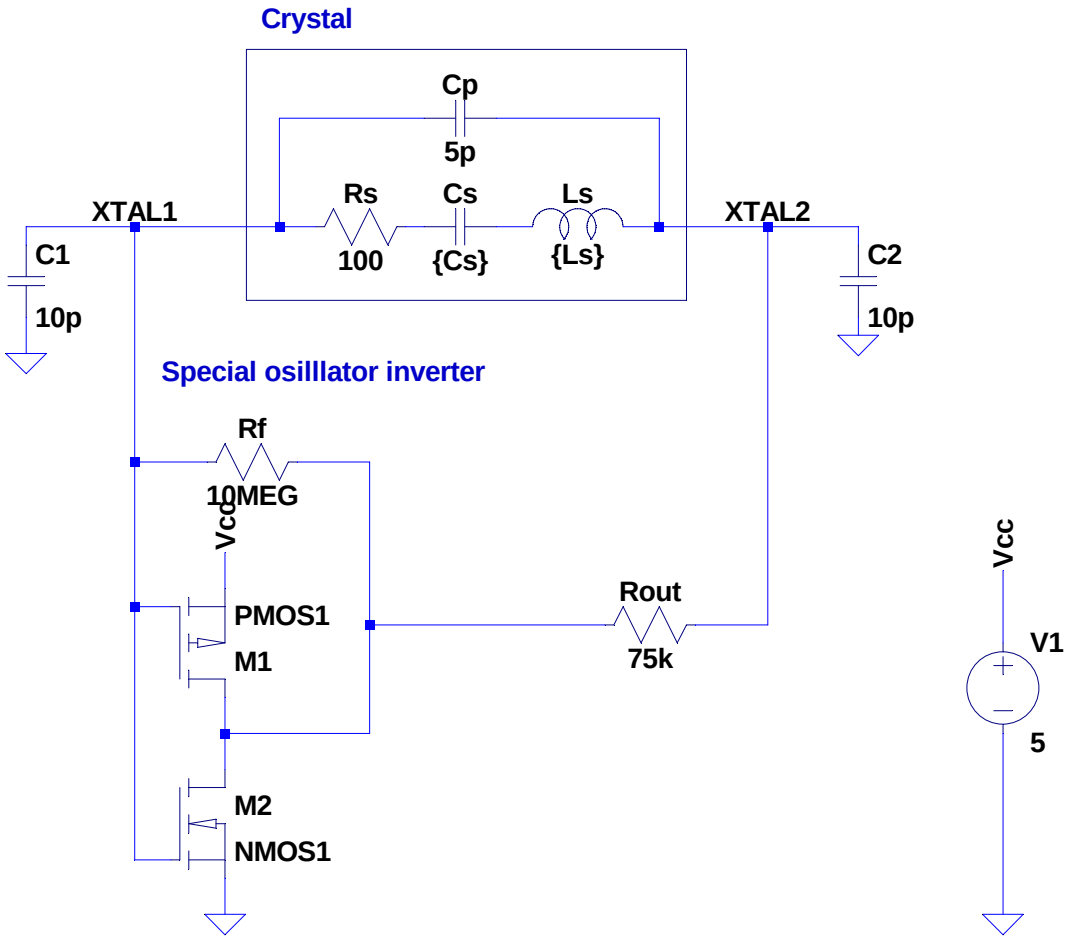
.TRAN 20ns 20ms 0.0199972 20ns UIC

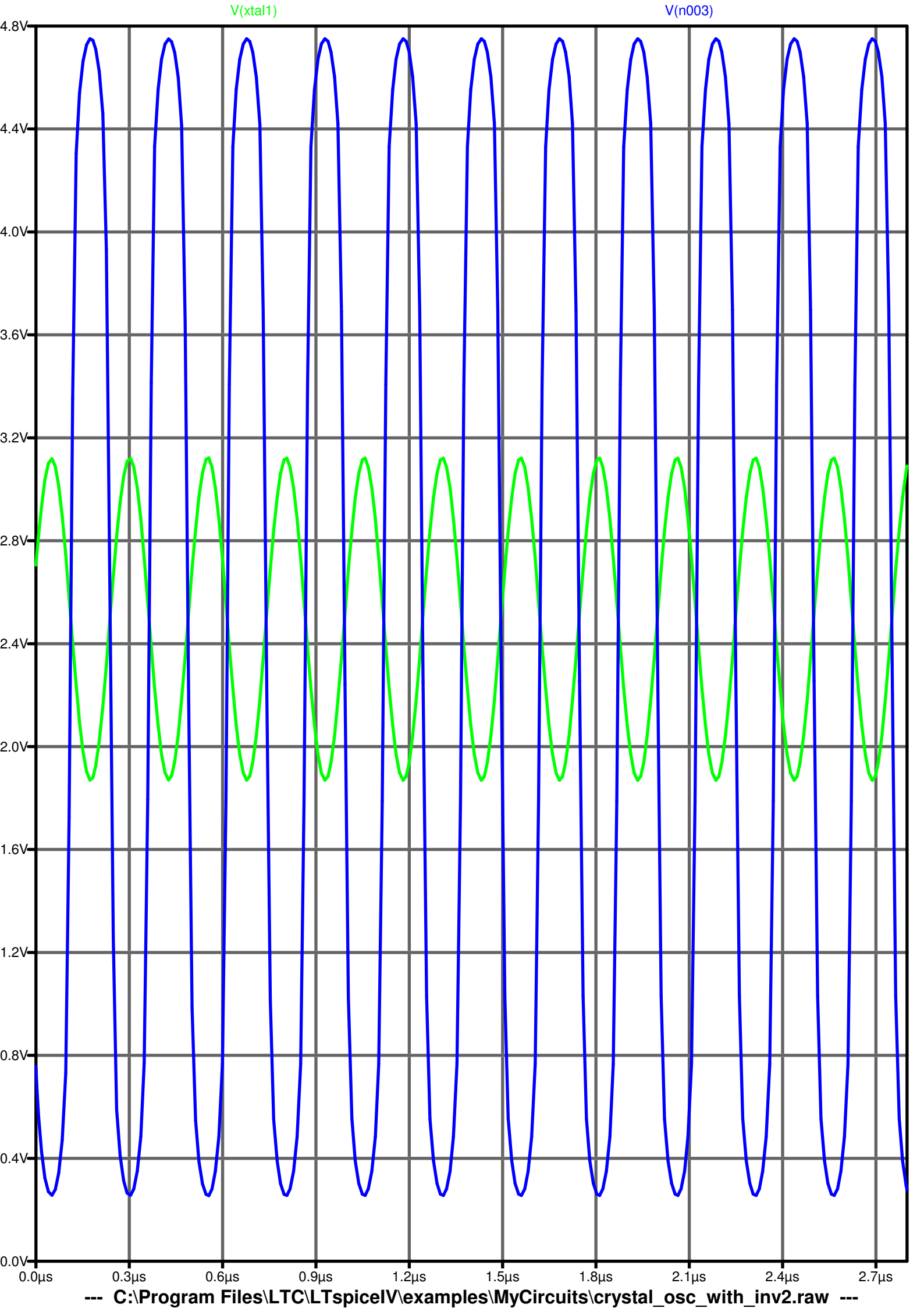
.PARAM f0=4e6Hz
.PARAM Ls=f0/1e9
.PARAM Cs=1/((2*pi*f0)**2*Ls)

.PARAM Ls=105.5m
.PARAM Cs=15e-15

.model PMOS1 pmos (kp=10u Vto=-1V lambda=0.04)
.model NMOS1 nmos (kp=20u Vto=+1V lambda=0.04)

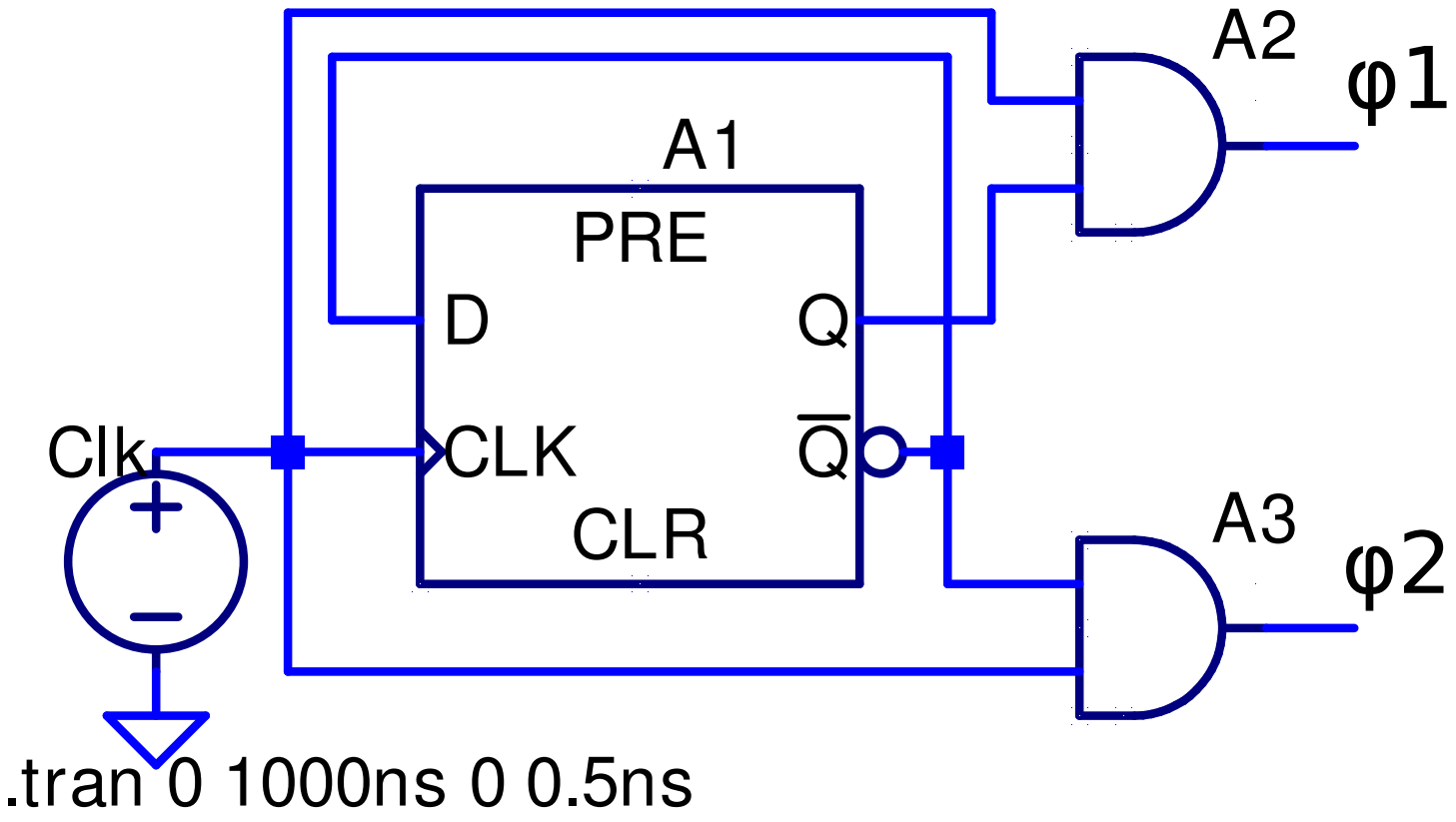
Crystal Oscillator With Inverter As Often Used In Digital ICs.
Rout and Rf must be added externally if a standard logic gate is used.

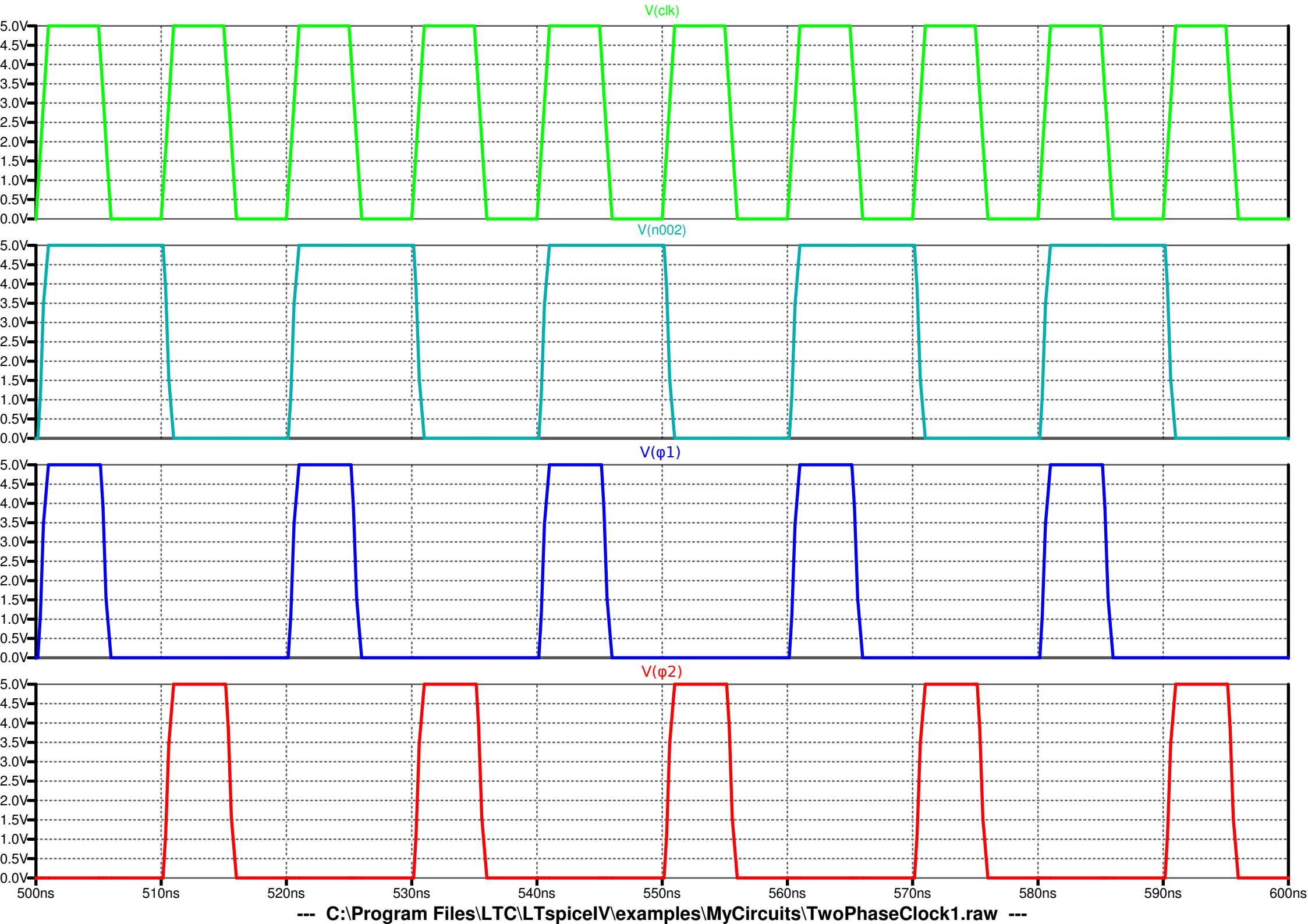



V(xtal1)
V(n003)
4.8V
4.4V
4.0V
3.6V
3.2V
2.8V
2.4V
2.0V
1.6V
1.2V
0.8V
0.4V
0.0V
0.0µs
0.3µs
0.6µs
0.9µs
1.2µs
1.5µs
1.8µs
2.1µs
2.4µs
2.7µs
--- C:\Program Files\LTC\LTspiceIV\examples\MyCircuits\crystal_osc_with_inv2.raw ---

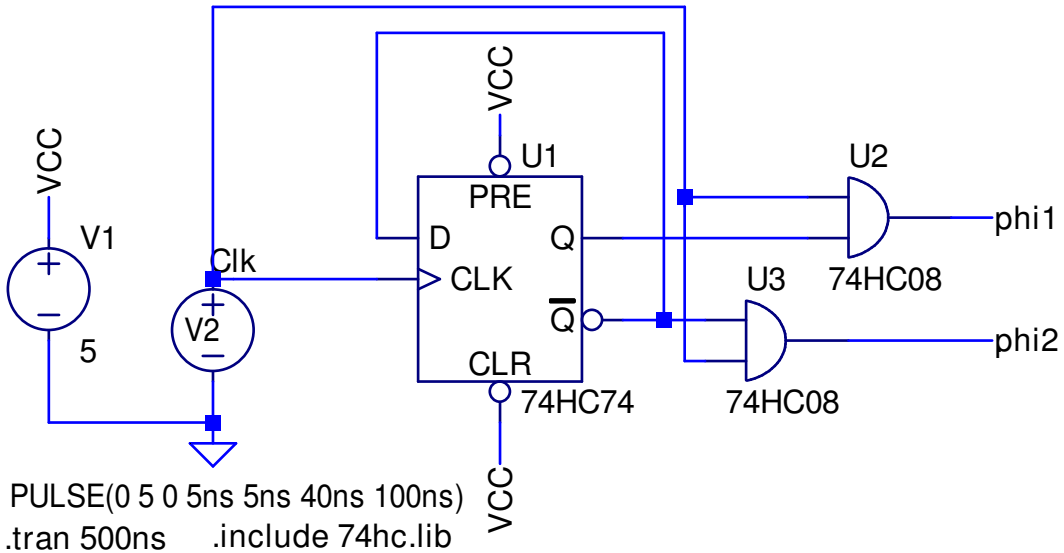
Two Phase non-overlapping Clock

Το κύκλωμα αυτό έχει παροδικό σφάλμα (glitch)

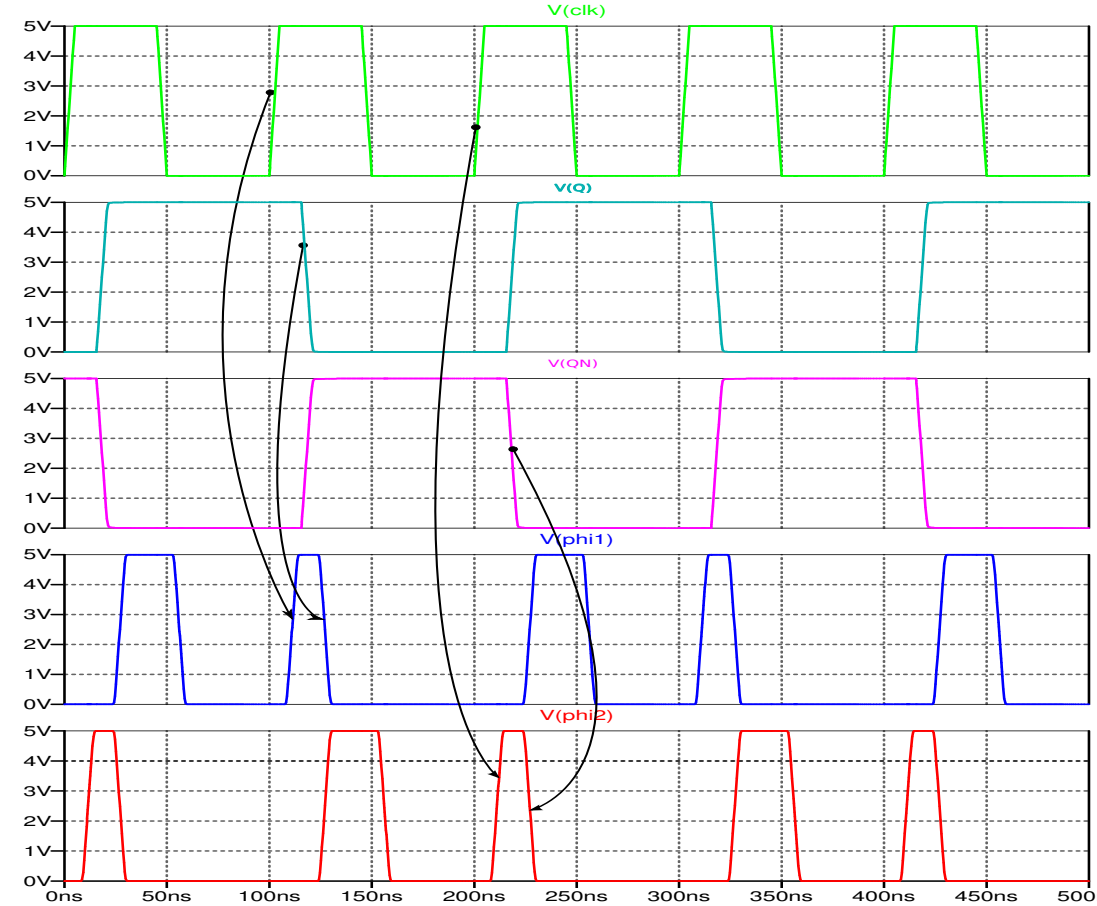



V(clk)
V(n002)
V(φ1)
V(φ2)
5.0V
4.5V
4.0V
3.5V
3.0V
2.5V
2.0V
1.5V
1.0V
0.5V
0.0V
500ns
510ns
520ns
530ns
540ns
550ns
560ns
570ns
580ns
590ns
600ns
--- C:\Program Files\LTC\LTspiceIV\examples\MyCircuits\TwoPhaseClock1.raw ---

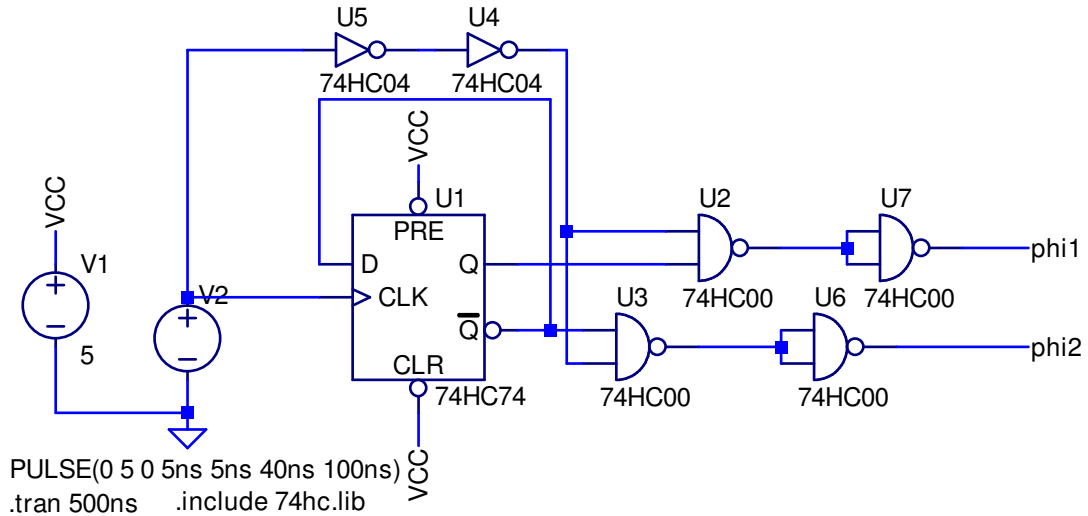
Two Phase non-overlapping Clock

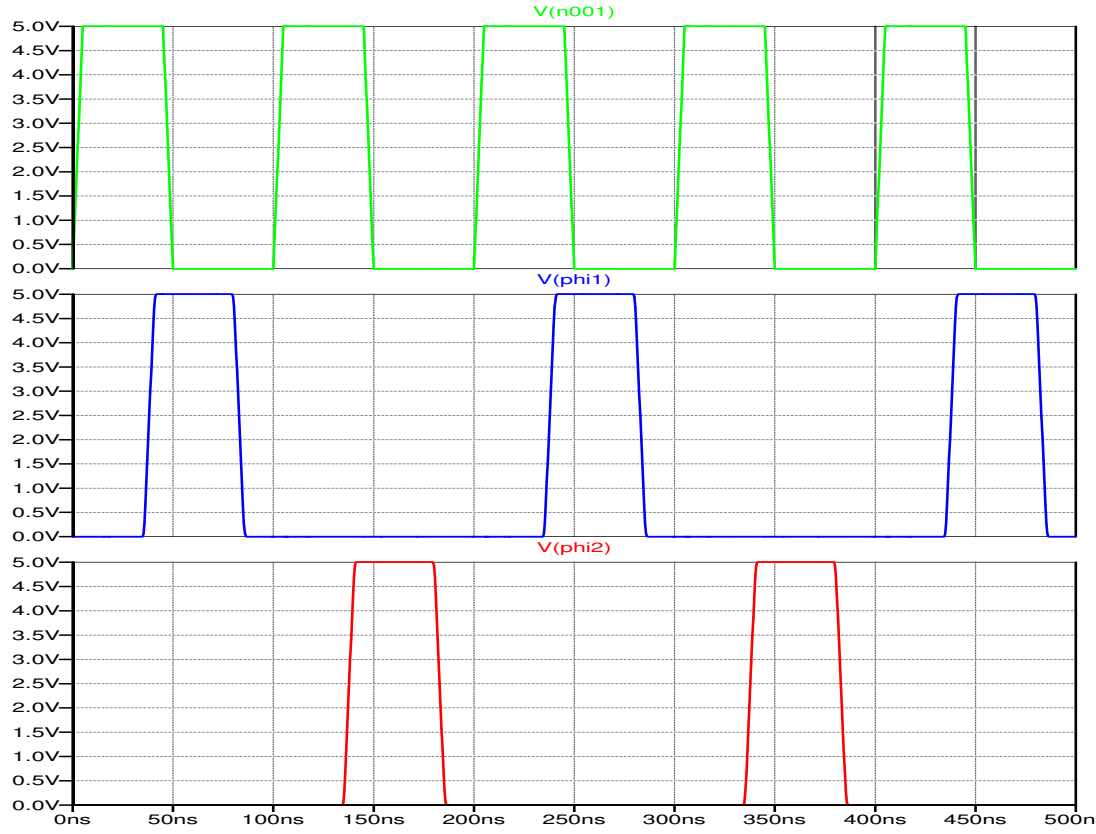


Το κύκλωμα αυτό έχει παροδικό σφάλμα (glitch)



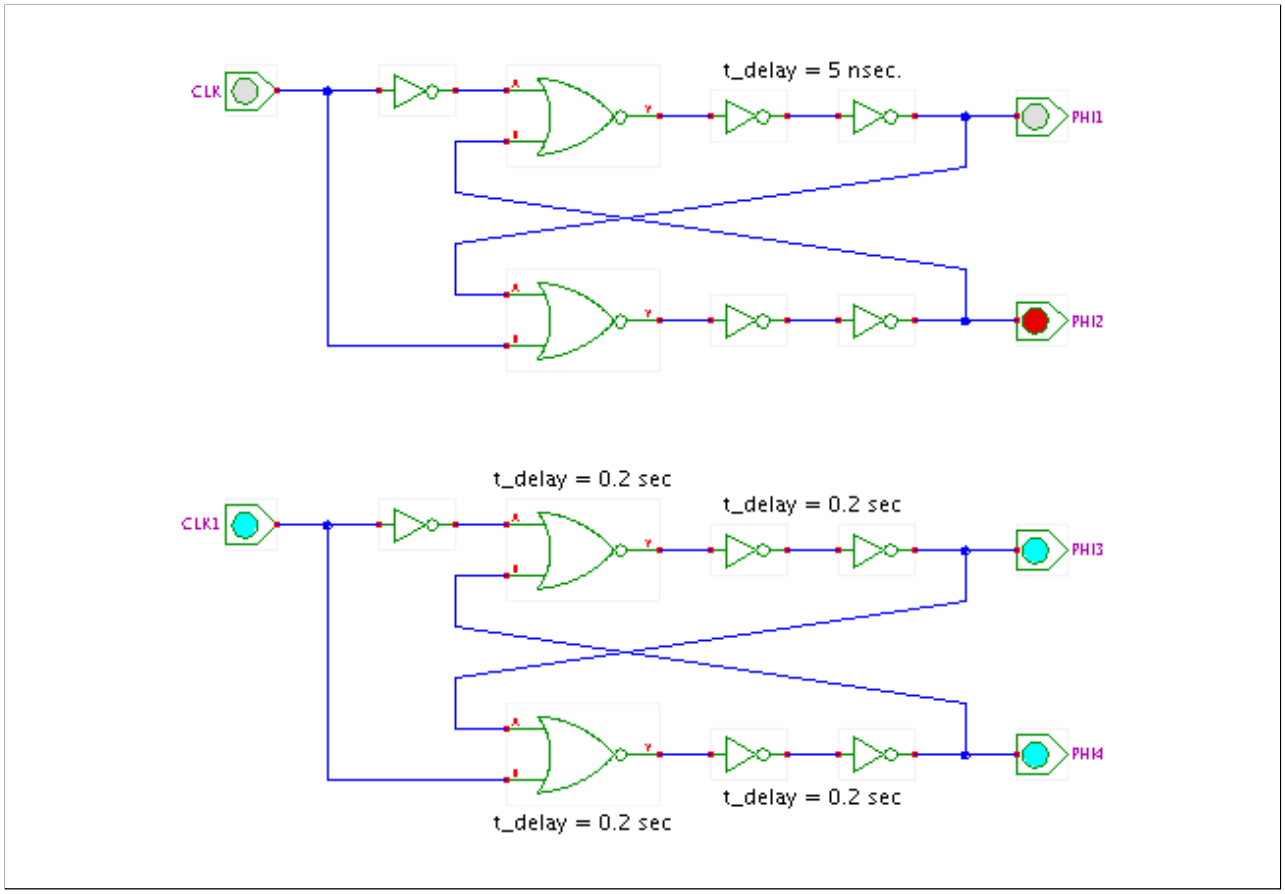
Two Phase non-overlapping Clock





Hades 2-phase clock generator

Two-phase non-overlapping clock generator

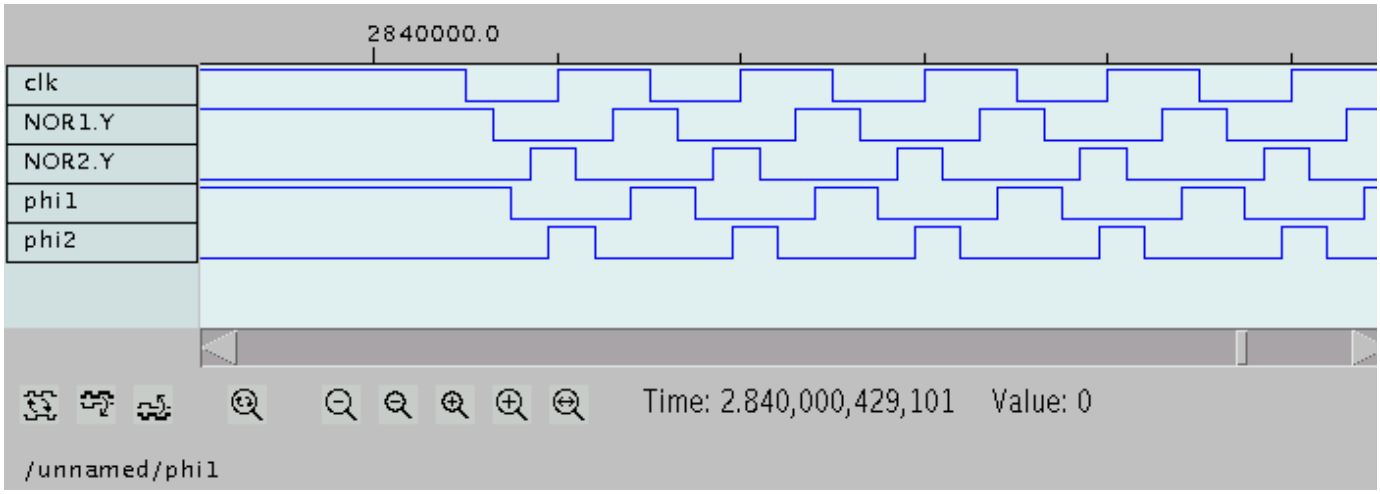


Description

This NOR-flipflop based circuit implements a **non-overlapping two-phase clock signal generator** and can be used to derive a two-phase clock signal from a single and possibly non-symmetrical clock signal. For an explanation of the circuit and a detailed discussion of circuit clock strategies see *N.H.E.Weste and K.Eshragian, Principles of CMOS design, 1993, section 5.5.10.*

While the upper circuit uses 'typical' gate-delays in the nanosecond range, the bottom circuit uses slowed-down gate-delays of 0.2 seconds per gate. This should make it easier to observe the idea behind the circuit.

A few example waveforms to illustrate the timing:



Run the applet | Run the editor (via Webstart)

Impressum | 24.11.06
http://tams-www.informatik.uni-hamburg.de/applets/hades/webdemos/12-gatedelay/40-tpcg/two-phase-clock-gen_print.html