

Computer-Based Project in VLSI Design Co 3/7

Design of logic gates in CMOS

This pamphlet gives a simple introduction to the principles of operation of CMOS logic gates, and attempts to point out some of the performance trade-offs that have to be addressed in their design. It also introduces the ring oscillator, which forms the heart of the design element of this project.

The CMOS inverter

The CMOS inverter or NOT gate is the simplest of the many boolean function that can be implemented using MOS transistors, but it serves to establish the fundamental principles which are common to all types. Our aim is to represent the logical values **1** and **0** by some electrical quantity and devise circuits that will allow us to manipulate these in the same way that boolean functions transform the logical values. It is conventional to represent logic **1** with a high voltage, and logic **0** with a low voltage. Over some forty years of evolution of transistorised logic, the industry has settled on standard values of 5 volts and 0 volts for this purpose. It should be realised that there is nothing particularly magical about these values, and indeed, within the last few years there has been a progressive shift to lower voltages: 3.3 V, 3 V, 1.8 V ...

In CMOS we use *enhancement mode* MOS (metal-oxide-silicon) FETs. These devices have no conductive channel (i.e. they are OFF) until the gate-source potential exceeds a specific value, the threshold voltage V_T . When the channel is fully formed, the device is said to be ON. The table below summarises the dependence of these devices on the gate-source voltage V_{gs} . The threshold voltages given are typical.

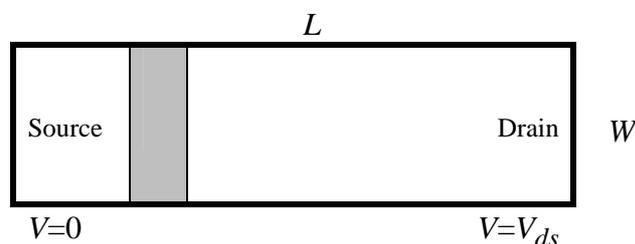
Type	V_T	$V_{gs} = 0$	$V_{gs} = 5$	$V_{gs} = -5$
n-type	+1v	OFF	ON	OFF
p-type	-1v	OFF	OFF	ON

The availability of *complementary* devices - that is, p-type and n-type - is one of the distinctive features of this technology. It is important to remember that the current equations for the p-type device involve inequalities of the opposite sign to those for the n-type device. In the n-type device, conduction occurs when p-type majority carriers are *repelled* from the vicinity of the channel, requiring that the gate be positive with respect to the source in order to switch the device ON. In the conductive p-type device, n-type carriers are repelled, requiring that the gate be negative relative to the source to turn the device ON. Thus a p-type device is turned OFF by a high gate voltage which turns an n-type device ON, and the converse holds for a low gate voltage. This complementary operation is of course the key to CMOS circuit design.

A simple model

The static characteristics of the CMOS inverter have been covered in Part IA and in third-year courses, and their derivation will not be repeated here. The theory of the devices themselves, and the approaches to fabrication required to make them work as fast as possible have already been covered in a Part IB Elective course, which many of you will have attended. In this project we shall use industry-standard simulation tools to model the behaviour of the devices and the circuits in which they operate. However, in order to be able to have an understanding of the results we observe, we shall develop a very simple approximate model for the conductance G of the MOS transistor channel, from which we can deduce many aspects of logic gate performance.

Consider the simple rectangular MOS transistor channel of length and width L and W respectively, shown below. Suppose the mobility of carriers within the material is μ .



Assume that the gate voltage $V_g = V_{dd}$, so that the device is conductive.

Let the charge density in the shaded element near the source be Q per unit length.

$$\text{Then } Q = C_{ox} V W$$

where V is the excess voltage on the gate, and is given by $V = V_{dd} - V_T$.

If V_T is much smaller than V_{dd} , we can approximate this:

$$Q = C_{ox} V_{dd} W$$

The current in the channel, I is given by:

$$I = \mu Q E$$

If we may assume that E is invariant along the device, we can write:

$$I \approx \mu Q V_{ds} / L = \mu C_{ox} V_{ds} V_{dd} W / L$$

Hence the conductance $G = I/V_{ds}$ is given by:

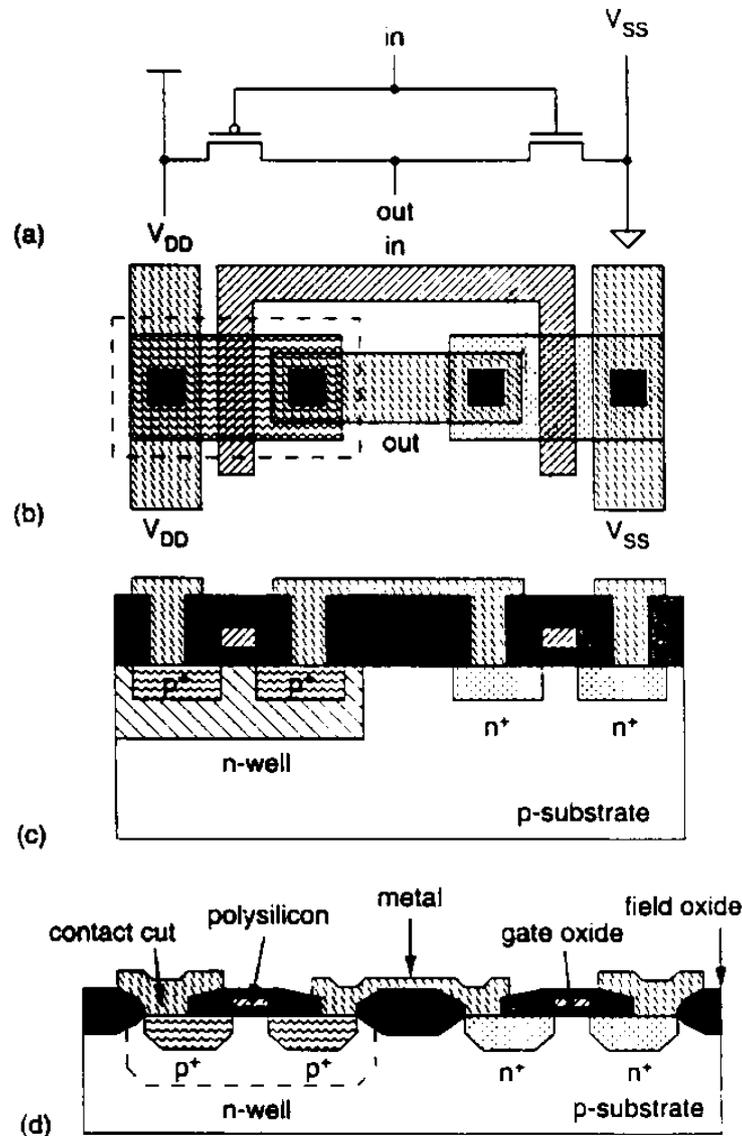
$$G = \mu C_{ox} V_{dd} W / L .$$

The result just obtained can be interpreted as follows. The channel conductance G (with the gate potential above threshold) is proportional to the factor μC_{ox} , often referred to as the *process gain factor*, and denoted by K . This is determined by material and manufacturing considerations, and is not under the direct control of the circuit designer. G is also proportional to the supply voltage V_{dd} , within the limits of the approximation, and - most importantly from the perspective of the designer - to a purely geometric factor W/L , often known as the *aspect ratio* of the MOS transistor.

Layout and performance of logic gates

We shall now use the simple result deduced above to predict some of the characteristics of the logic inverter, and shall then explore how this treatment can be extended to more complicated gates.

The circuit schematic and layout of a CMOS inverter are shown below.



The transistor channel is formed by the superposition of the polysilicon gate electrode over the thin gate oxide which is grown on the surface of the lightly doped semiconductor itself. The source and drain are formed from highly doped p- or n-type material created using ion implantation.

It can be seen that the channel width W is determined by the width of the thin oxide region (that is, the distance it extends into the plane of the paper), while the channel length L is actually defined by the *width* of the stripe of polysilicon which extends across the channel. Both these quantities are directly under the control of the designer, and are in fact the primary means by which the designer determines the performance of each element of the system.

A logic circuit may be characterised by three key parameters:

- Physical size
- Power consumption
- Speed of operation

An ideal circuit will have minimum size, fastest speed and the lowest power demand. It turns out, however, that it is not possible to optimise all characteristics simultaneously, and the designer must weigh his requirements characteristics and balance the competing constraints to suit the application.

Physical size

This is proportional to the product of W and L . Minimising the values of W and L clearly give the smallest size. However, this approach is limited by the ability of the manufacturing process to produce reliable devices at the smallest dimensions. Like any manufacturing process, it is subject to tolerances which set a lower limit to the size that can be achieved. For example, the current minimum channel length presently available from commercial fabrication facilities is about 0.4 - 0.8 μm (though many designs require greater channel lengths). The designer is thus constrained by a set of rules (*Design Rules*) which govern the dimensions of the various structures he specifies.

Speed

When the input to the inverter switches between 0 and 1 or *vice versa*, the output terminal is expected to change state correspondingly. The metal and semiconductor elements connected to the output all possess capacitance to the substrate (and to all other conductors, to be pedantic). We shall regard these separate contributions as a single *lumped* capacitance C connected between the output and ground. This capacitance must be charged and discharged through the conductive channel of one or other transistor: the discharge to 0V via the n-type channel, and the charge-up to V_{dd} via the p-type channel. Although there are other factors that determine the delay experienced by a signal in travelling between the input of a logic gate and its output, under most circumstances it is the processes just outlined that dominate. It is possible to develop accurate equations relating the drain current to gate and drain potentials. These can be integrated to determine how long the charge or discharge processes take. However, in the interests of simplicity we shall regard the system as a set of switched conductances which control the charge/discharge dynamics, and we shall assume that

the conductances are constant, and defined by the simple conductance model given earlier.

Using classical RC network theory, we can now write down expressions for the delay for *rising* and *falling* edges at the output terminal. We may also assume that the gate potential changes abruptly, so that the channels also change state abruptly between conductive and non-conductive. A convenient pair of expressions, which are well borne out by experiment, are as follows:

$$\text{Delay time for rising edge} \approx 3 C / G_p \qquad \text{Delay time for falling edge} \approx 3 C / G_n$$

As expected, the smaller the value of C , the shorter the delays and the faster the circuit. Considering the expressions for G , it is clear that the higher the supply voltage V_{dd} , the more rapidly it will switch.

From the IC designer's point of view, the larger the value of W , the higher the conductance and hence also the faster the circuit. There is a subtlety here which we must not overlook. The capacitance C is to some extent dependent on the values of L and W , since the gate itself and the depletion regions forming the drains of the transistors may make a sizable contribution to the total parasitic capacitance at the output terminal. Increasing W may not necessarily produce the speedup that the simple expression suggests.

There is yet a further complication. Although the values of V_{dd} and C_{ox} are usually the same for both n- and p-type transistors, the mobility μ differs significantly between the types. Typically, the ratio μ_n / μ_p may be two or more. This means that for given values of W and L , the conductances of n- and p-channel devices will be different by the same factor. An inverter using identically-sized transistors would therefore have different delays for rising and falling edges. In real logic circuits, asymmetric delays of this kind are a disadvantage, and measures must be taken to equalise them. For the inverter this is easily achieved by adjusting the ratio W/L for each device such that $\mu W/L$ is constant. To keep the devices compact, we may use the minimum permissible value of L , and scale W . This results in the value of W for the p-channel being greater than that for the n-channel device by a factor μ_n / μ_p .

Power

Power is the rate of consumption of energy. One of the great virtues of the CMOS inverter, is that provided the input terminal is held either at logic **0** or logic **1**, only one transistor is conductive, the other being non-conductive. As a result there is never under these circumstances a direct pathway for current to flow from V_{dd} to ground. This means in its turn that the power consumed is negligible. These considerations have led to the use of CMOS in many applications where lower power operation is essential.

This convenient model for power dissipation in CMOS is in fact too simple. Charge does actually flow from V_{dd} to ground every time the output terminal switches from logic **1** to logic **0**. While the output terminal is high, the parasitic capacitance charges up to a potential V_{dd} , and holds a charge $C V_{dd}$. When the output terminal changes state, this charge is conducted to ground through the n-channel transistor. A fixed packet of charge $C V_{dd}$ is thus transferred from power supply to ground each time the input of the gate changes state from **1** to **0** and back again. If the gate is driven by a

periodic waveform of frequency f , we can identify a flow of current from V_{dd} to ground, via the parasitic capacitance, whose average value I_{av} is given by:

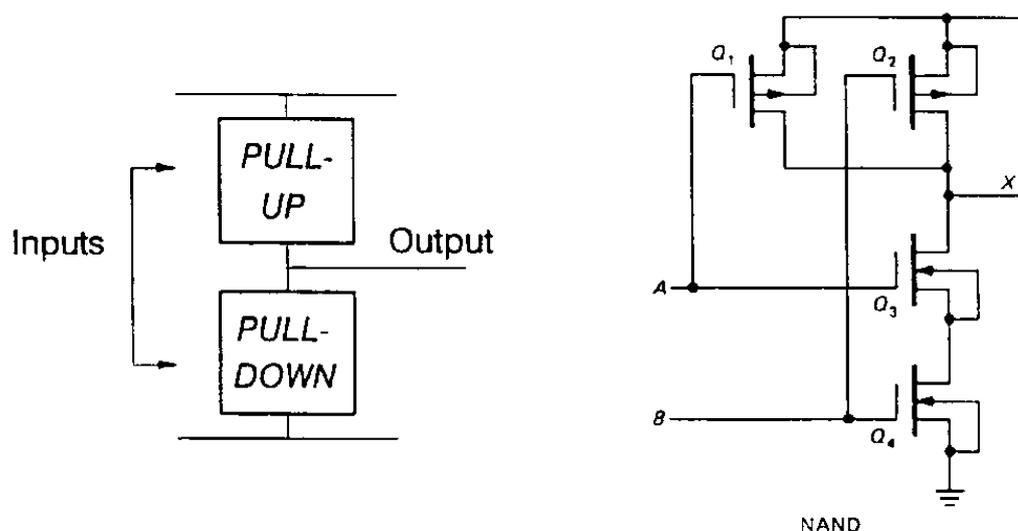
$$I_{av} = f C V_{dd}$$

Hence the average power consumption by this single gate is $V I_{av}$, or $f C V^2$. We can apply this kind of reasoning to entire CMOS integrated circuits, provided we know the parasitic capacitances associated with each gate output. Estimating the frequency of operation of every individual gate may be rather more difficult. As a first approximation it may be satisfactory to assume that each gate is driven at some fraction (say, about one half) of the externally-supplied clock signal connected to the circuit, on the basis that gates will be clocked at frequencies ranging from DC right up to the clock frequency, and in roughly equal quantities. Where more information is available about signal frequencies, a more accurate dissipation figure can be determined.

This form of *dynamic* loss is the principal mechanism by which dissipation of power occurs in CMOS. At sufficiently high frequencies, the amount of heat produced can be enough to require special measures to disperse it (the Pentium and DEC Alpha chip are well known illustrations, and dissipate several watts because they run at 100MHz+). This is a further reason to try and minimise the parasitic capacitances. Note that the power dissipation is also proportional to V^2 . Hence increasing V_{dd} in an attempt to improve switching speed has the disadvantage of significantly increasing the dynamic power dissipation. Many new, high speed devices are designed to run at 3 V or even less for this very reason.

More complicated gates

The arguments just developed for the CMOS inverter can be extended to other more complicated combinational gates. Any arbitrary combinational gate in CMOS can be expressed as a *pull-up* network (consisting of p-channel MOS transistors, and connected to V_{dd}) and a *pull-down* network (consisting of n-channel transistors, and connected to ground), as in the diagram below.



In CMOS each input signal controls a device in the pull-up network and another in the pull-down network. The networks are designed in such a way that **either** the pull-up network connects the output to V_{dd} , **or** the pull-down connects the output terminal to ground. The networks must never be simultaneously conductive.

For example, the transistor schematic for a two-input NAND gate is also shown in the diagram, comprising two n-channel transistors in series (pull-down) and two p-channel transistors in parallel (pull-up).

Let us consider now the issue of delay in more complicated gates. Firstly, the existence of additional MOS transistors connected to the output terminal increases the parasitic capacitance observed there. This is an inescapable fact. Secondly, we see that our simple model for charge/discharge of this capacitance through a single p-type or n-type device has to be extended. In the NAND gate, the discharge takes place through two n-type devices Q3 and Q4 in series. Of course, *both* devices must be conductive to allow this to take place. However, the charging current may flow through Q1, Q2, or Q1 **and** Q2, since these devices are parallel-connected. We need to reconsider how to determine the overall conductance of the pull-up and pull-down networks when these may consist of arbitrarily complicated groupings of transistors in series/parallel. A reasonable way of calculating effective conductances for series-connected devices is to use Ohm's Law. Since the 'resistance' of a transistor is $1/G$ and is therefore proportional to L/W , we can argue that for a string of n series-connected transistors:

$$\left[\frac{L}{W} \right]_{\text{effective}} = \left[\frac{L}{W} \right] + \dots + \left[\frac{L}{W} \right]_n$$

This is actually not such a good approximation as one might expect, because it fails to take account of the observed non-linear nature of channel conductance, which is dependent on the source and drain voltages as well as on the gate potential. In other words, the conductance of each of two (physically identical) transistors with identical gate voltages, connected in series, may *not* be identical, since they are quite likely to have different source and drain voltages. This is a fairly fundamental limitation of our simple conductance model. However, we shall ignore this problem for now, as to do otherwise would tend to obscure the important fundamental principles we wish to emphasise.

With parallel-connected transistors, we have observed that the conduction may take place through one or more devices, according to the state of the inputs. When estimating the delay times for a gate, it is normal to use worst-case assumptions. These may then be used to determine worst-case figures for the delay through an entire system, a conservative approach to design. Hence the effective conductance for a parallel set of devices may reasonably be taken as the conductance of just one of those devices, representing the worst case.

Applying this rationale to the 2-input NAND gate, we see that the worst-case pull-up conductance is given by:

$$\mu_p C_{ox} V_{dd} \left[W_p / L_p \right]$$

with only one p-channel device conducting, while the pull-down conductance is:

$$0.5\mu_n C_{ox} V_{dd} [W_n/L_n]$$

since *two* n-channel devices are in series. In order to equalise the *worst-case* rising and falling delays, we must set these equal. It follows from this that:

$$[W_p/L_p] = 0.5 \frac{\mu_n}{\mu_p} [W_n/L_n]$$

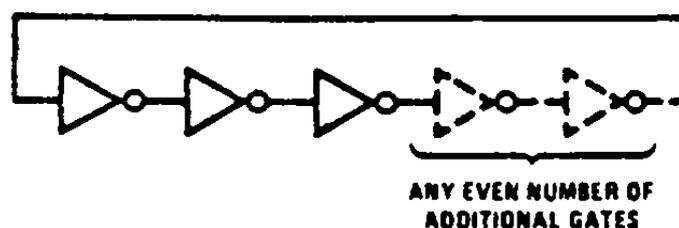
A further issue which we shall mention but not develop fully, is how we might attempt to normalise the rising and falling delays of gates with multiple inputs to those of the inverter. There may be worthwhile advantages in a family of logic gates with roughly uniform delay, and it is apparent that we can at least improve the situation by choosing $[W_n/L_n]$ for multi-input gates so that the worst-case conductance in the pull-down network is similar in magnitude to that for the inverter. This in its turn affects the aspect ratio chosen for the p-channel devices.

We should note that selecting the transistor dimensions in this way cannot in general match rising and falling delay times for *every* possible input transition. For the 2-input NAND, the delay for a rising input will be a factor of two faster than the worst case if both inputs are switched simultaneously from logic **1** to logic **0** rather than just one.

The Ring Oscillator

The ring oscillator is at least conceptually one of the simplest forms of oscillating circuit. It has a number of technical limitations that render it less than ideal for general-purpose oscillatory circuits, but it has some fascinating features which are of great value to the integrated circuit design engineer.

An ideal ring oscillator consists of a homogeneous chain of identical inverting gates connected in cascade - that is, with the output of each stage connected to the input of the next. The output of the final gate is connected back to the input of the first, giving a closed ring-like structure. See the diagram below.



Ring oscillator - the simple implementation

Note that the circuit shows an *odd* total number of gates. The simplest way to understand how such an oscillator works is to imagine that any convenient point - say, the input to gate number 1 - attains the value logic **0** at time 0. Since each gate inverts, the output from gate 1 will become logic **1** a short time after (dependent on the delay imposed on the signal as it passes through gate 1). The output from gate 2 will become logic **0** a further gate delay later, and so on. It is possible to visualise a transition (logic **1** to logic **0**) racing around the ring, being inverted and delayed as it progresses. When the transition reaches the output from the *n*th gate (that is, the input to gate 1), it has undergone an *odd* number of inversions, and therefore is of the opposite polarity, i.e. logic **0** to logic **1**, and arrives at time t_1 determined by the delay through *n* gates. The inverted transition races around the ring and reaches the starting point after a further delay t_2 , and is now of the same polarity as the initial transition. It is clear that so long as the signals assume regular logic levels, the circuit has no stable state, and will continue to oscillate.

To determine the oscillation period, it is necessary to know the delay through each stage. From above we see that during a complete cycle of oscillation a transition must travel round the ring two complete times. If all stages have identical delay τ both for rising and falling edges, it follows that the period of oscillation is $2n\tau$. Hence the frequency of oscillation is:

$$f = 1 / 2n\tau$$

Since there is such a close link between τ and the frequency of oscillation, the ring oscillator has considerable value in the design of integrated circuits. It provides a convenient way in which the delay of a simple gate can be measured to a high degree of accuracy with comparatively simple instrumentation. This information is vital in order to provide a check on the fabrication process, and also to determine proper numerical models for the devices. The highest speed gates currently available have delays of a few tens of picoseconds. To measure this kind of delay directly is extremely difficult. Moreover, even if instrumentation were available with a suitable response, the capacitive load it would impose on the gate under test would grossly distort the measurement.

The solution is to build a ring oscillator comprising a suitably large number of identical gates. With the ring oscillating, the period of oscillation may be measured and the expression above used to determine the individual gate delay. Any combinational gate capable of performing an inversion may be used in place of the inverter shown. Hence this technique can be applied to NAND, NOR or even XOR and XNOR gates to measure their performance. Virtually all integrated circuits nowadays have a small area set aside for a *parametric 'drop-in'* - manufacturers' jargon for a small piece of circuitry - typically including a ring oscillator - inserted specifically to allow verification that the processing has complied with expectations.

Examples - solutions should be included in your First Interim Report

1. Referring to the section above: **Design of Logic Gates in CMOS**, show how to draw the pull-up and pull-down networks in such a way as to produce CMOS gates which perform following logic functions

$$(a) Y = \overline{A.(B + C + D)}$$

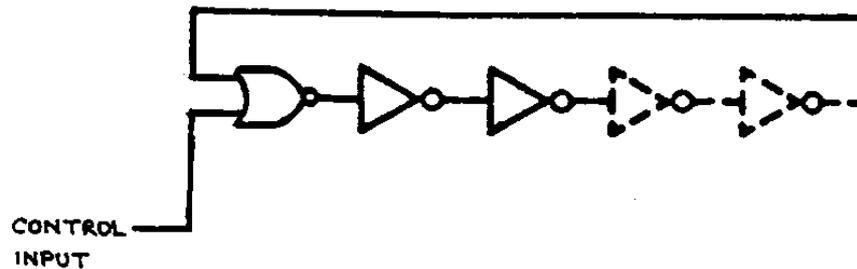
$$(b) Y = \overline{A + (B.C.D)}$$

2. How must the transistors be dimensioned to achieve equal *worst-case* delays for rising and falling edges with the gates (a) and (b) considered above. Assume that μ_n / μ_p is 2.5, and that *minimum* allowed values for L and W are $2\mu\text{m}$ and $4\mu\text{m}$ respectively.
3. Comment on the ring oscillator method for determining the delay through a gate. What, if any, are the advantages and disadvantages of this approach to delay measurement compared with a more direct technique?
4. Determine the frequency of oscillation of a ring oscillator if each of the gates used has delay τ_r for rising edges and τ_f for falling edges.
5. How might you vary the frequency of a ring oscillator? Hint: consider the simple expressions for delay and transistor conductance developed earlier in this sheet.
6. Is it possible for a ring oscillator to oscillate at any other basic frequency than that given by the expression: $f = 1 / 2n\tau$? Explain your reasoning.
7. Should it be possible for a ring oscillator with an even number of gates oscillate? Under what limited conditions might oscillation be detectable?

A practical ring oscillator circuit

The simple ring oscillator circuit outlined above well illustrates the principles of operation, but it is not a particularly practical arrangement. For convenience, we shall wish to have some way of switching on or off the train of oscillations, and we shall require more than one output so we can observe the phase relationships between them.

To achieve this we require at least one gate to have a second input (i.e. NAND or NOR) so that an external signal can be applied, as in the improved circuit below.



Ring oscillator - a more practical design

To maintain homogeneity in the ring, and for convenience, we shall use 2-input NOR gates for every stage of the ring; where only a single input is required, we shall simply wire together the two inputs A and B, giving the functionality of an inverter. A library part - NOR2 - will be used in the first instance to provide us with a feel for the way the circuit behaves, but in the final design we shall create a mask layout to implement our own version of the 2 input NOR gate, and investigate how its performance compares with that of Mietec's design.

The required specification for the ring oscillator part of the design is provided in the following section.

