

## LAB #3: DIGITAL AND ANALOG CMOS APPLICATIONS

Updated Dec.23, 2002.

### Objective:

To investigate a variety of CMOS applications, both digital and analog. To compare Pspice simulations with experimental observations.

### Components:

1 × CD4007UB MOSFET Array, 1 × CD4020 Binary Counter; capacitors: 1 × 10 nF, 2 × 0.1 μF; resistors: 8 × 10 kΩ, 9 × 20 kΩ, 3 × 100 kΩ, 1 × 1.0 MΩ (all 5%, ¼ W).

### Instrumentation:

A bench power supply, a triangular signal generator, a pulse generator, a digital multi-meter, and a dual-trace oscilloscope.

## PART I – THEORETICAL BACKGROUND

CMOS technology offers a number of important advantages:

- When sitting in a particular logic state, a digital circuit dissipates *zero power*, as there is at least one transistor in the circuit path from the supply to ground which is operating in cutoff.
- Digital circuits offer *excellent noise margins*, particularly the buffered types.
- MOSFETs exhibit virtually *infinite input impedance*, making loading much easier to handle.
- Digital circuits swing their outputs all the way up to  $V_{DD}$  and all the way down to  $V_{SS}$ , a feature known as *rail-to-rail output swing*, which results in highly predictable digital levels.
- As exemplified by the availability of both *digital* and *analog* gates, CMOS technology lends itself to the implementation of both *digital* and *analog* functions on the same chip, making this technology ideal for *mixed-mode* ICs.

As we embark upon the present laboratory, we recall the following important MOSFET properties:

- When operated in the ohmic region with a sufficiently small voltage across its channel, a MOSFET acts as a *voltage-controlled resistor*. The channel resistance of an *n*MOSFET is

$$r_{DSn} = \frac{1}{k_n (v_{GS} - V_{tn})} \quad (1a)$$

and that of a *p*MOSFET is

$$r_{SDp} = \frac{1}{k_p (v_{SG} - |V_{tp}|)} \quad (1b)$$

where  $k$  is the *device transconductance parameter*, and  $V_t$  is the *threshold voltage*.

- When operated in the pinchoff region under small-signal conditions, a MOSFET, whether of the *n* or the *p* type, provides the *transconductance*

$$g_m = \sqrt{2kI_D} \quad (2a)$$

with the parallel *output resistance*

$$r_o = 1/\lambda I_D \quad (2b)$$

where  $I_D$  is the *DC current* through the device, and  $\lambda$  is the *channel-length modulation factor*. Equations (1) and (2) form the basis of analog applications of CMOS.

### The CD4007UB MOSFET Array:

The purpose of this laboratory is to investigate a variety of CMOS applications using the transistors of the CD4007UB MOSFET Array that you have characterized in Lab #1. For convenience, the pinout diagram of this IC is repeated in Fig. 1. Beware that all wiring tips and experimental precautions stressed in Lab #1 still hold! If you damage the particular CD4007 sample you are working with, you'll need to characterize a new sample in order to ensure consistency between measurements and simulations. MOSFETs belonging to the *same* IC sample are indeed matched to a good degree, but no matching can be expected between *different* IC samples.

To prevent possible damage to the gate oxide by the inadvertent application of excessively large input voltages from the outside, *all inputs are protected* by an internal network of diode clamps. This network, deliberately omitted from the diagrams of Fig. 1 for simplicity, is shown in Fig. 2 for the input (Pin 6) of transistors  $M_1$  and  $M_2$  (a similar network exists for Pins 3 and 10). It is apparent that diodes  $D_2$  and  $D_3$  restrict the actual voltage  $v_G$  appearing right at the gates of  $M_1$  and  $M_2$  within the range

$$(V_{SS} - 0.7 \text{ V}) \leq v_G \leq (V_{DD} + 0.7 \text{ V})$$

and that  $R$ , typically in the range of a few  $k\Omega$ , limits the current through these diodes. Moreover, diode  $D_1$  clamps the voltage of Pin 6 at  $V_{DD} + 0.7 \text{ V}$ . Should a careless user force a voltage higher than  $V_{DD} + 0.7 \text{ V}$  at this pin, diode  $D_1$  may blow out. So, be careful!

Henceforth, steps shall be identified by letters as follows: **C** for calculations, **M** for measurements, **P** for pre-lab preparations, and **S** for Spice simulations. As usual, all data must be expressed in the form  $X \pm \Delta X$ .

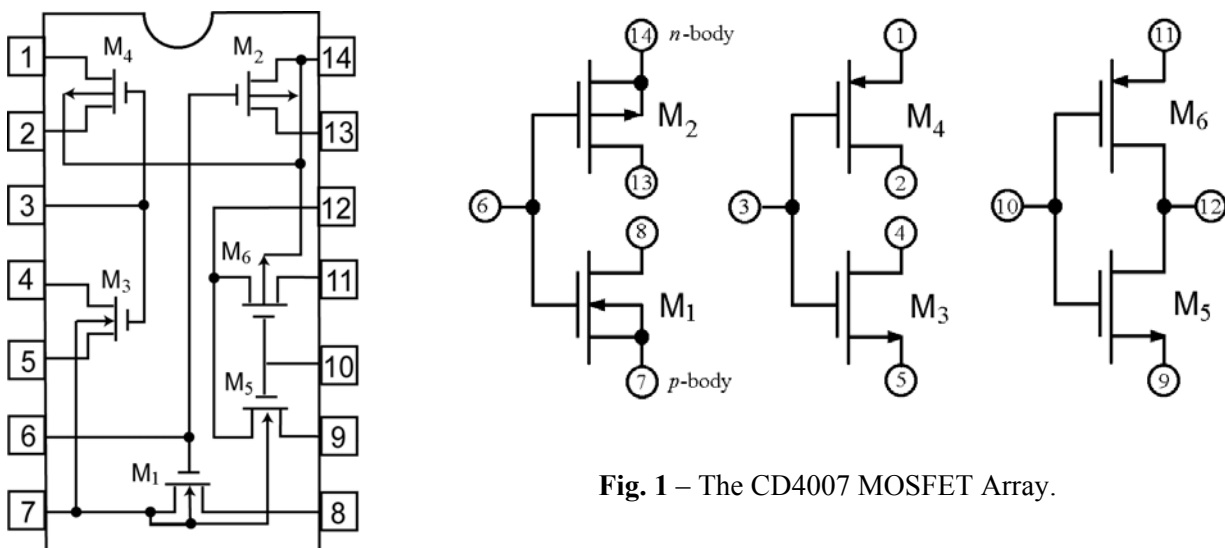


Fig. 1 – The CD4007 MOSFET Array.

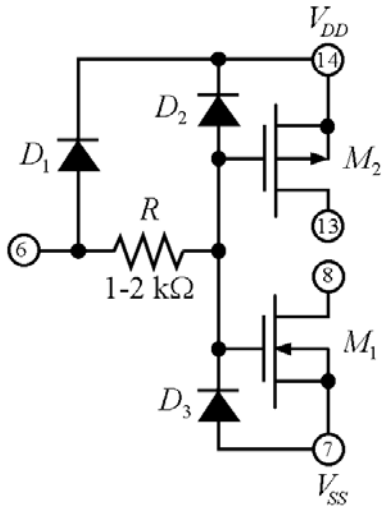


Fig. 2 – Internal diode protective network.

## PART II – EXPERIMENTAL PART

### Astable Multivibrator:

Also called a *free-running multivibrator*, this circuit acts like a flip flop that alternates spontaneously (that is, without the need of any external input) between its two possible states, in effect functioning as a square-wave generator. The timing is established by an external network, which in its simplest form consists of a capacitor  $C$  and a resistor  $R$ . A more stable and accurate square-wave generator, especially for precise clock applications, uses a quartz crystal as the timing element. Figure 3 shows a popular free-running multivibrator, which uses two CMOS inverters, along with a resistor  $R$  and a capacitor  $C$  to establish the timing. Note that the circuit has no external input, but it provides an output  $v_o$ , which is a square wave. What matters to the user is the *frequency*  $f_0$  of this square wave.

The circuit can readily be simulated via PSpice. In the PSpice example of Fig. 4, the inverters are implemented using homebrew MOSFETs called, respectively, 453nMOSFET and 453pMOSFET. Both devices were created by renaming and suitably editing the *PSpice Models* of two MOSFETs available in the PSpice Library. This has been done first by clicking the device to select it, then by clicking **Edit** → **PSpice Model** to change the values of its parameters. The model statements for the two devices are as follows:

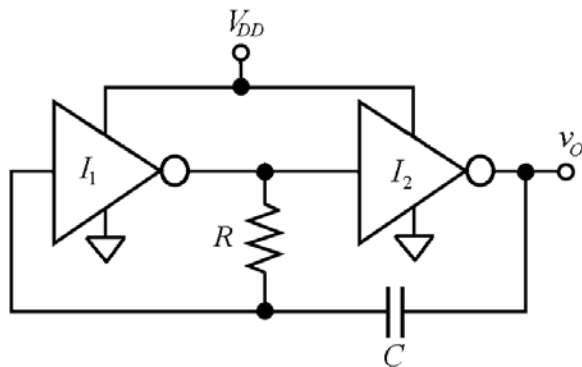
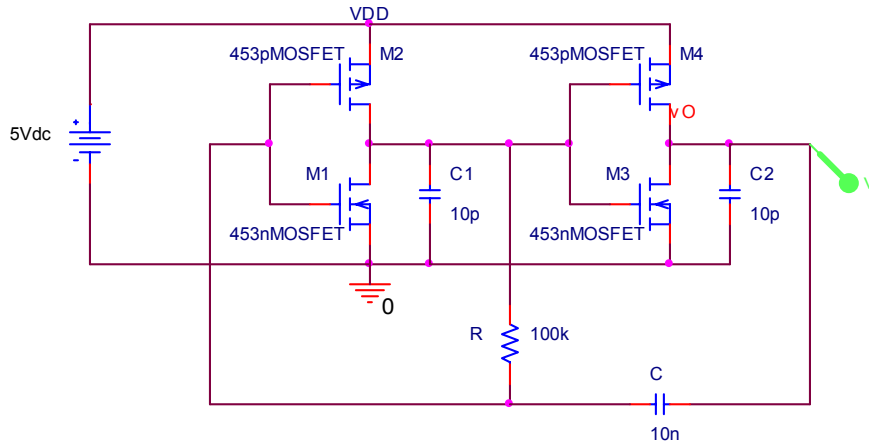


Fig. 3 – CMOS astable multi-vibrator



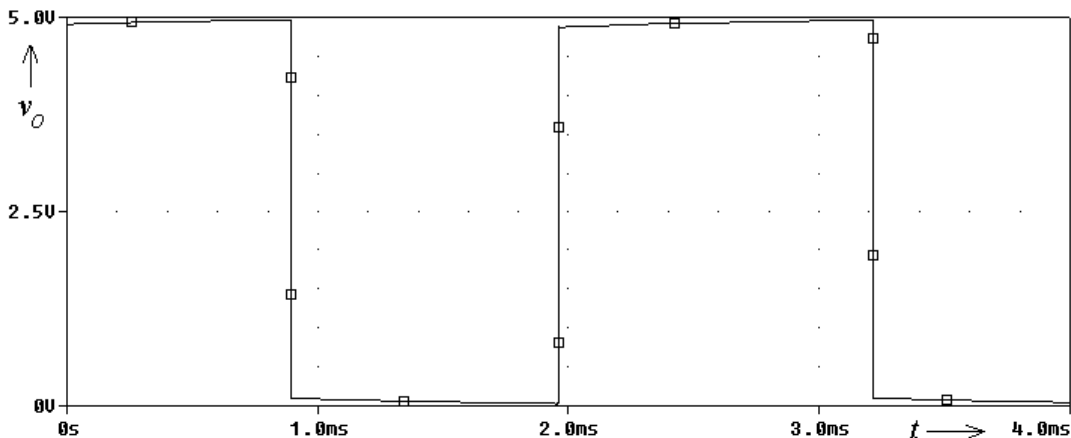
**Fig. 4** – PSpice circuit to simulate a CMOS astable multi-vibrator.

```
.model 453nMOSFET NMOS(W=4u L=1u kp=50u Vt0=1 lambda=0.01)
.model 453pMOSFET PMOS(W=8u L=1u kp=20u Vt0=-1.5 lambda=0.02)
```

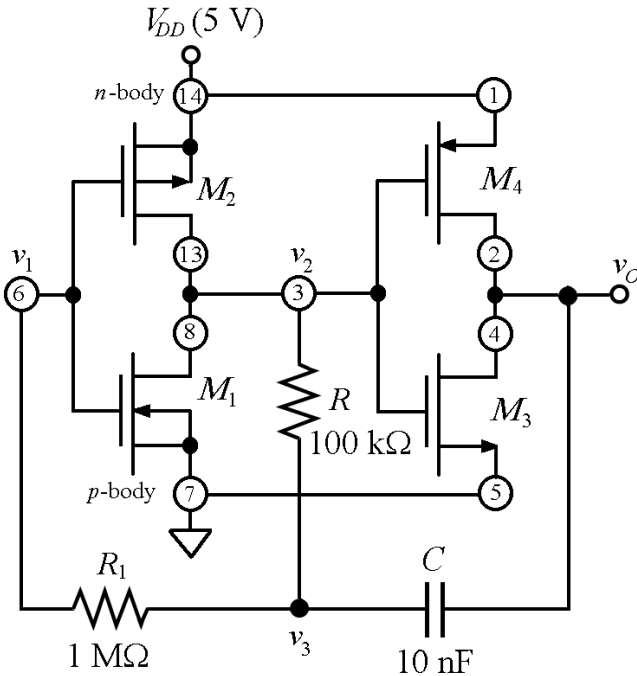
Note the inclusion of capacitors  $C_1$  and  $C_2$  at the outputs of the inverters to account for their nonzero propagation delays. Moreover,  $C_1$  has been initialized to 5 V.

Figure 5 shows the output waveform, where we note that the duty cycle is not necessarily 50%, due to the fact that the trip voltage  $V_T$  is not necessarily  $\frac{1}{2}V_{DD}$ . You can simulate the above circuit on your own by going to <http://online.sfsu.edu/~sfranco/CoursesAndLabs/Labs/453Labs.html>. Once there, click on **PSpice Examples**, and then follow the instructions contained in the **Readme** file.

Figure 6 shows an experimental implementation using the MOSFETs of the CD4007 Array. The purpose of  $R_1$  ( $R_1 \gg R$ ) is to prevent the diode clamp  $D_1$  associated with Pin 6 (refer to Fig. 2) from significantly loading the voltage waveform  $v_3$  at the node common to  $R$  and  $C$ .



**Fig. 5** – Astable multivibrator output



**Fig. 6** – Astable multivibrator using the CD4007 MOSFET array..

**PS1:** Using the *n*MOSFET and *p*MOSFET values of  $V_{t0}$ ,  $k$ ,  $\lambda$ , and the inverter's equivalent output capacitance  $C_{eq}$  determined in Lab #1, perform a PSpice simulation of the multivibrator of Fig. 6. Display a few periods of all significant waveforms in your circuit ( $v_1$ ,  $v_2$ ,  $v_3$ , and  $v_O$ ), and use the cursor facility of PSpice to determine the period  $T$  of oscillation. Perform your simulation (a) first with the circuit as shown, and (b) then with a diode  $D_1$  between Pin 6 (anode) and Pin 14 (cathode) to simulate the clamping action of  $D_1$  discussed in connection with Fig. 2 (for this purpose, you can use the diode D1N4148 available in PSpices' library.) Compare the waveforms at Pin 6, and discuss how significant the impact of  $D_1$  upon  $T$  is.

**PC2:** Based on the waveforms observed in Step PS1, provide a word description of how the circuit works. Next, derive an expression for the period  $T$  of oscillation in terms of  $R$ ,  $C$ ,  $V_{DD}$ , and the trip voltage  $V_T$ . Finally, use this expression to calculate  $T$ , compare with the value found in Step PS1, and account for any differences.

**M3:** With power off, assemble the circuit of Fig. 6. Then, apply power and use the oscilloscope to observe and record all significant waveforms. Measure the period  $T$ , and calculate the frequency of oscillation as  $f_0 = 1/T$ . Compare with the values of  $T$  of the previous steps, account for any differences.

**M4:** In many applications it is desirable that  $f_0$  be independent of  $V_{DD}$ . We wish to find how well this holds for our specific circuit. Thus, while monitoring  $V_{DD}$  with the digital voltmeter and  $v_O$  with the oscilloscope, observe the effect of increasing  $V_{DD}$  from 5 V to 10 V upon  $f_0$ . Express the power-supply sensitivity  $\Delta f_0 / \Delta V_{DD}$  of your oscillator in Hz/V. Explain what might cause  $f_0$  to depend upon  $V_{DD}$ .

### Monostable Multivibrator:

Also called a *one-shot*, this circuit acts like a flip flop that prefers to sit in one of its states, typically the state  $Q = 0$ . Applying an external trigger pulse forces the circuit to change state ( $Q = 1$ ), but only for a predetermined time interval  $T$ , after which the circuit returns spontaneously to its stable state ( $Q = 0$ ). Aply called *timeout interval*,  $T$  is again set by an external capacitor  $C$  and resistor  $R$ . Figure 7 shows

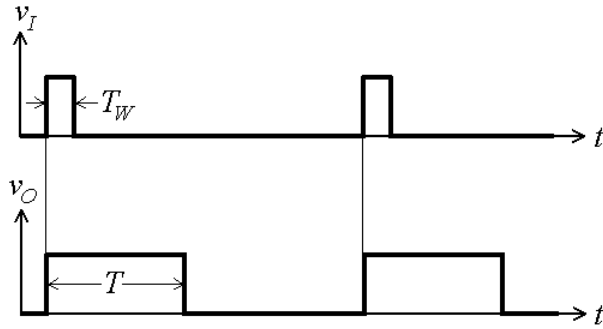


Fig. 7 – One-shot timing.

typical input and output waveforms. The input trigger is usually a narrow pulse of width  $T_W \ll T$ . If  $T_W$  happens to exceed the intended value of  $T$ , the actual timeout interval may no longer be  $T$ , depending on how the circuit is implemented. This uncertainty can be overcome by designing the one-shot so that it responds only to the leading *edge* of the input trigger pulse, regardless of its width.

Figure 8 shows a simple one-shot implementation based on two CMOS gates (the NOR of Fig. 9a of Lab #2, and a plain inverter), along with a resistor  $R$  and a capacitor  $C$  to establish the value of  $T$ . Note that now the circuit, in addition to the output  $v_O$ , has also provision for a pulse input  $v_I$  for its triggering.

**PS5:** Using the CMOS parameters of Step PS1, perform a PSpice simulation of the one-shot of Fig. 8. At the input of the inverter use a diode  $D_1$  between Pin 10 (anode) and Pin 11 (cathode) to simulate the clamping action of  $D_1$  discussed in connection with Fig. 2 (for this purpose, you can use the diode D1N4148 available in PSpice’s library). As input trigger use a train of pulses alternating between 0V and 5 V, with width  $T_W \cong 1 \mu\text{s}$ , and spaced apart by a couple of milliseconds. Display all significant waveforms in your circuit ( $v_I$ ,  $v_1$ ,  $v_2$ , and  $v_O$ ) for several input trigger pulses. Hence, use the cursor facility of PSpice to determine the timeout interval  $T$ .

**PC6:** Based on the waveforms observed in Step PS5, provide a word description of how the circuit works. Next, derive an expression for the timeout interval  $T$  in terms of  $R$ ,  $C$ ,  $V_{DD}$ , and the trip voltage  $V_T$ . Finally, use this expression to calculate  $T$ , compare with the value found in Step PS5, and account for any differences.

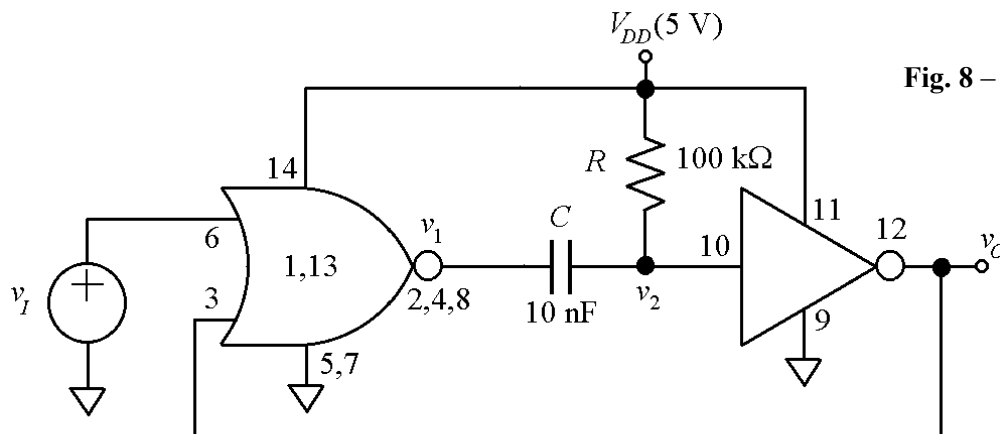
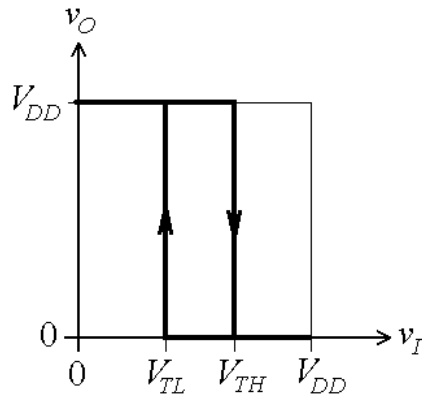


Fig. 8 – CMOS one-shot



**Fig. 9** – VTC of a Schmitt trigger

**M7:** Assemble the circuit of Fig. 8 with power off, and adjust the pulse generator for a pulse train of the type of Step PS5. Next, apply power, connect the signal generator to your circuit, and use the oscilloscope to observe and record all relevant waveforms. Finally, measure the timeout interval  $T$ , compare with the values of  $T$  of the previous steps, and account for any differences.

**Exploration:** What happens if you lengthen  $T_w$  until  $T_w > T$ ? Explain in terms of gate behavior.

### Schmitt Trigger:

With reference to the VTC of an inverter, we observe that as we sweep  $v_I$  from 0 V to 5 V, we expect  $v_O$  to make a single transition from 5 V to 0 V, and vice versa. However, if  $v_I$  happens to be a slow-varying and noisy signal, when  $v_I$  approaches  $V_T$  its noise component gets amplified by the high gain exhibited there by the inverter, and this may cause  $v_O$  to undergo multiple transitions, or *bounces*. Bouncing is generally intolerable, and an effective way to eliminate it is by having the VTC exhibit *hysteresis*. As illustrated in Fig. 9, when  $v_I$  is swept from LOW to HIGH the gate trips at  $v_I = V_{TH}$ , whereas when  $v_I$  is swept from HIGH to LOW the gate trips at  $v_I = V_{TL}$ . You can readily convince yourself that as long as the noise component of  $v_I$  has a peak-to-peak amplitude less than the hysteresis with  $W = V_{TH} - V_{TL}$ , no multiple bounces will occur.

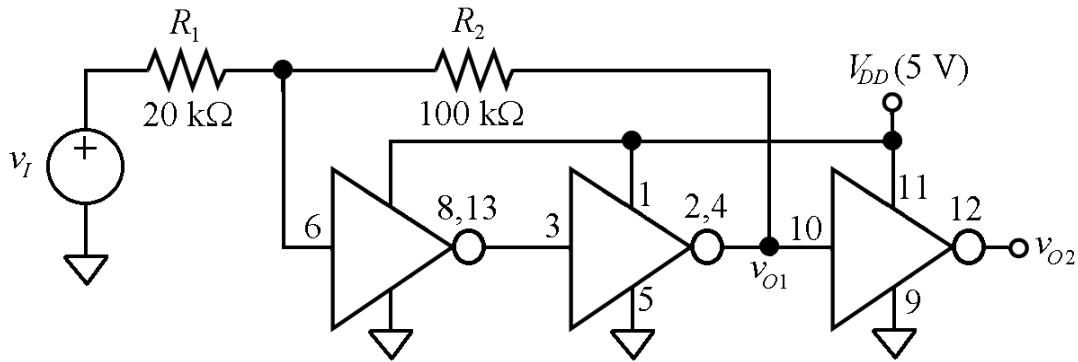
To achieve hysteresis, the tripping point must be made to depend on the output through *positive feedback*. Circuits with this type of feedback are generally known as *Schmitt Triggers*. In CMOS gates specifically designed to exhibit hysteresis, positive feedback is achieved *internally* using the channel resistances of suitable MOSFETs in the feedback path. Two popular products are the CD4093 Quad 2-Input NAND Schmitt Triggers and the CD40106 Hex Schmitt Triggers.

In the present laboratory we are going to investigate this principle using *external* resistances instead, in the manner depicted in Fig. 10. Depending on whether we consider the output to be  $v_{O1}$  or  $v_{O2}$ , we have a *noninverting* or an *inverting* Schmitt trigger, respectively.

**PS8:** Use PSpice to display *both* the inverting and the noninverting VTC of the Schmitt Trigger of Fig. 10. (For the MOSFETs, use the parameters of Step PS1; moreover, to walk your circuit through its entire hysteresis cycle, you must sweep  $v_I$  first from 0 V to 5 V, and then from 5 V back to 0 V.) Finally, use the cursor facility of PSpice to determine the values of  $V_{TL}$  and  $V_{TH}$ .

**PS9:** Use PSpice to display all voltage waveforms in the circuit of Fig. 10 for the case in which  $v_I$  is a 1-kHz triangular waveform alternating between 0 V and 5 V.

**C10:** Based on your observations of Steps PS8 and PS9, provide a word description of how the circuit of Fig. 10 works. Next, derive expressions for  $V_{TL}$  and  $V_{TH}$  in terms of  $R_1$ ,  $R_2$ ,  $V_{DD}$ , and the trip voltage  $V_T$ .



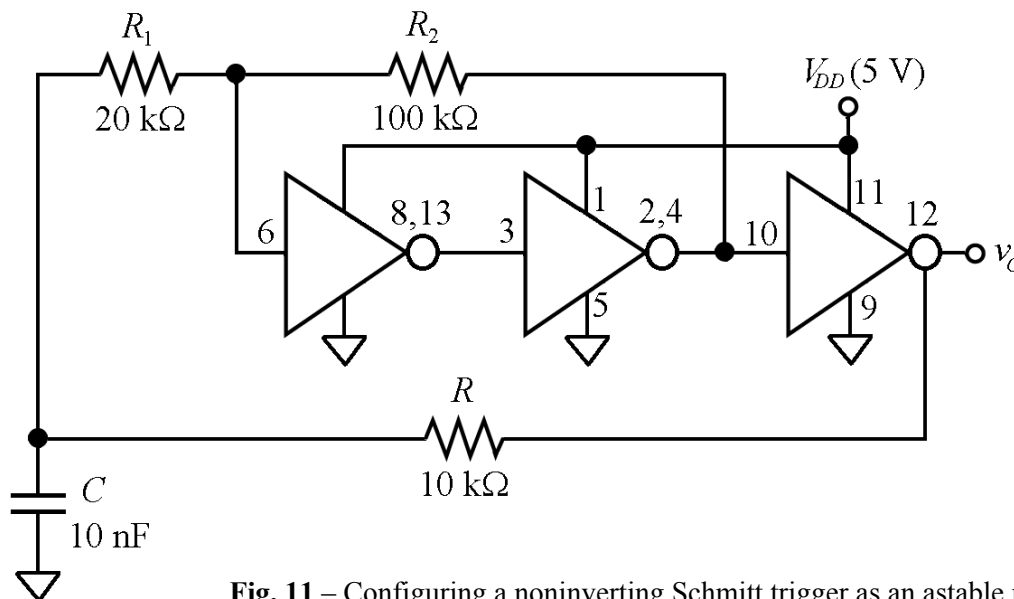
**Fig. 10** – CMOS Schmitt trigger implementation using external resistances.

Finally, use these expressions to calculate  $V_{TL}$  and  $V_{TH}$ , compare with the values found in Step PS1, and account for any differences.

**M11:** With power off, assemble the circuit of Fig. 10. Next, observe its VTC experimentally with the oscilloscope, determine the values of  $V_{TL}$  and  $V_{TH}$ , compare with those of Step PC10, and justify any differences.

### The Schmitt Trigger as an Astable Multivibrator:

Schmitt triggers find application not only as debouncers, but also in a variety of other situations. An example is offered by Fig. 11, where the inverting-type Schmitt Trigger has been configured to operate as an astable multivibrator in conjunction with an external  $R$ - $C$  network to establish the frequency  $f_0$  of oscillation.



**Fig. 11** – Configuring a noninverting Schmitt trigger as an astable multivibrator.

**M12:** With power off, assemble the circuit of Fig. 11. Then, apply power and use the oscilloscope to observe and record all significant waveforms. Measure the period  $T$  of oscillation, and calculate the frequency of oscillation as  $f_0 = 1/T$ .

**C13:** Based on the waveforms observed in Step M12, provide a word description of how the circuit works. Next, derive an expression for the period  $T$  in terms of  $R$ ,  $C$ ,  $V_{DD}$ ,  $V_{TL}$ , and  $V_{TH}$ . Finally, use this expression to calculate  $f_0$ , compare with the value found in Step M12, and account for any differences.

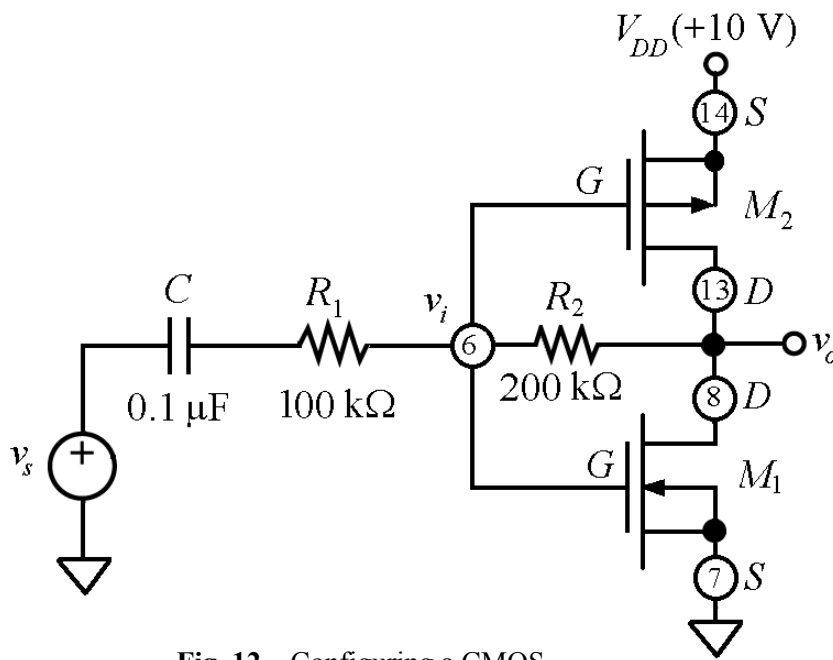
### Analog Applications:

The portion of a CMOS inverter's VTC centered around the trip voltage  $V_T$  is of great interest in *analog applications*. In this region, aptly called the *linear region*, both MOSFETS operate in *pinchoff*, and the slope  $a = dv_O/dv_I$  is *steepest*; clearly,  $a$  is the *voltage gain* of the circuit. Recall that because of differences in the parameters of the *n*MOSFETS and *p*MOSFETS,  $V_T$  is not necessarily halfway between  $V_{SS}$  and  $V_{DD}$ .

To operate the inverter as an amplifier, we need to bias it in the linear region. A simple technique is to use a *feedback resistance*  $R_2$  (see Fig. 12), which forces both the input and the output of the inverter to stabilize at a common DC bias voltage  $V_{BIAS}$  in the vicinity of  $V_{DD}/2$ . Since the inverter's input draws zero current,  $R_2$  can be chosen as large as desired. In Fig. 12 a second resistor is used, namely,  $R_1$ , to configure the inverter for operation as an *inverting amplifier* with a gain of  $-2$  V/V. The purpose of  $C$  is to provide AC coupling from source to amplifier.

**PC14:** Using the MOSFETS' *large signal* models, show that the input and output of the inverter of Fig. 12 will automatically stabilize at the common DC voltage  $V_O = V_I = V_{BIAS}$ , where

$$V_{BIAS} = \frac{V_m + \sqrt{k_p/k_n}(V_{DD} - |V_p|)}{1 + \sqrt{k_p/k_n}} \quad (3)$$



**Fig. 12** – Configuring a CMOS inverter as an *inverting amplifier*.

and that the *range of linear operation* is

$$(V_O - V_m) < v_O < (V_{O+} + |V_{tp}|) \quad (4)$$

Then, using the CMOS parameters of Step PS1, compute  $V_{BIAS}$  for  $V_{DD} = 10 \text{ V}$ , as well as the extremes of the linear range of operation.

**PC15:** Using the MOSFETs' *small signal* models, show that over the range of Eq. (4), the *open-loop small-signal gain* is

$$a = \frac{v_o}{v_i} = \frac{-(g_{mn} + g_{mp} - 1/R_f) \times (r_{on} // r_{op})}{1 + (r_{on} // r_{op})/R_f} \quad (5)$$

where subscripts  $n$  and  $p$  refer to the  $n$ MOSFET and  $p$ MOSFET, respectively. Then, using the CMOS parameters of Step PS1, along with Eq (2), compute the gain  $a$  for  $V_{DD} = 10 \text{ V}$  and  $R_2 = 200 \text{ k}\Omega$ .

**PC16:** Show that for a sufficiently high input frequency to make  $C$  act as a short compared to  $R_1$ , the *closed loop gain* is

$$A = \frac{v_o}{v_s} = \frac{-R_2/R_1}{1 + \frac{R_2/R_1}{a}} \quad (6)$$

Then, using the result of Syep PC15, compute the gain  $A$  for  $R_1 = 100 \text{ k}\Omega$  and  $R_2 = 200 \text{ k}\Omega$ .

**PS17:** Verify the results of Steps PC14 through PC16 via PSpice, and account for any differences.

**M18:** With power off, assemble the circuit of Fig. 12 (for  $R_2$  use  $2 \times 100 \text{ k}\Omega$  in series), and adjust the waveform generator so that  $v_s$  is a 10-kHz triangular wave with a peak-to-peak amplitude of 0.5 V. Keep leads short and bypass the 10-V supply to ground via a 0.1- $\mu\text{F}$  capacitor. Next, apply power, connect the generator to your amplifier, and while using Ch. 1 (set on AC) to monitor  $v_s$ , use Ch. 2 (also set on AC) to observe and record first  $v_o$  and then  $v_i$ , the input to the inverter itself. Hence, find the *open-loop* and the *closed-loop* gains  $a$  and  $A$  by taking the ratio of the peak-to-peak amplitude of  $v_o$  to that of  $v_i$  and  $v_s$ , respectively. Compare with the values predicted by Eqs. (5) and (6), and account for any discrepancies.

**M19:** Gradually increase the amplitude of  $v_s$  until  $v_o$  begins to exhibit significant distortion compared to a nearly-ideal triangular shape. Record both  $v_o$  and  $v_i$ , and justify their *distorted shapes* in terms of your findings of Steps PC14 and PS17.

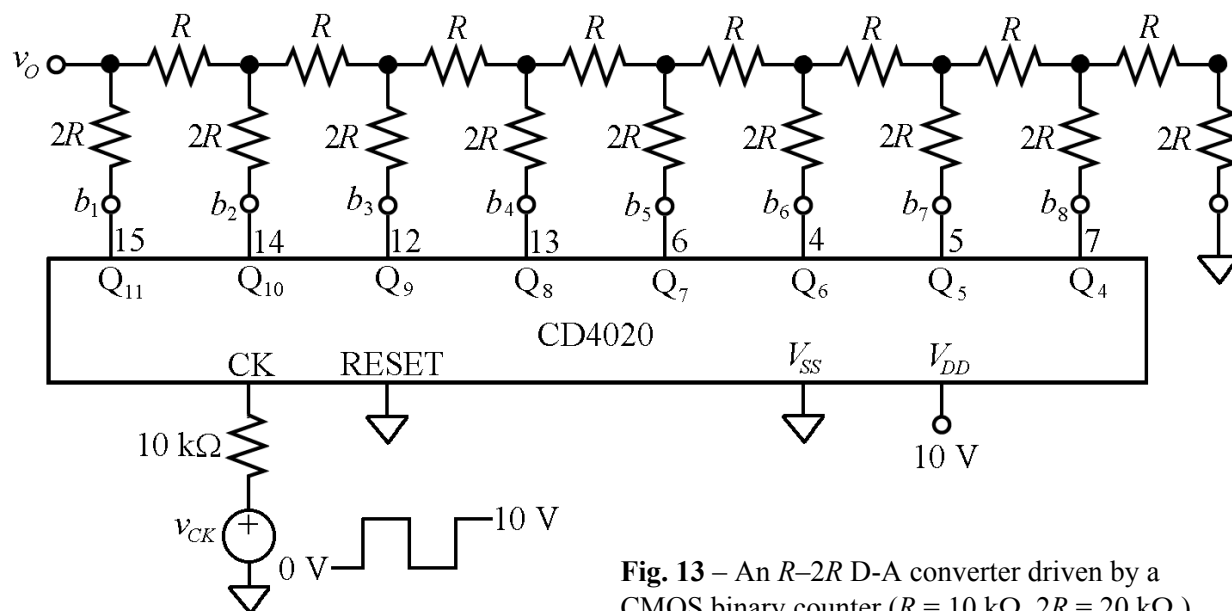
### Digital-to-Analog Converter:

A digital-to-analog (D/A) converter (DAC) accepts an  $n$ -bit *input word*  $b_1 b_2 b_3 \dots b_n$ , with  $b_1$  being the most significant bit (MSB) and  $b_n$  being the least significant bit (LSB), and produces *an analog output* of the type

$$v_O = V_{REF}(b_1 2^{-1} + b_2 2^{-2} + b_3 2^{-3} + \dots + b_n 2^{-n}) \quad (7)$$

where  $V_{REF}$  is a suitable scaling factor called the *reference voltage*, and bits  $b_1$  through  $b_n$  take on the

values of either 0 or 1. A popular way to implement a DAC is by using an  $n$ -leg terminated *R-2R ladder*, along with a set of  $n$  single-pole double-throw (SPDT) switches to connect the individual ladder legs ( $2R$ )



**Fig. 13** – An  $R$ - $2R$  D-A converter driven by a CMOS binary counter ( $R = 10\text{ k}\Omega$ ,  $2R = 20\text{ k}\Omega$ ).

either to  $0\text{ V}$  or to  $V_{REF}$ . MOSFETs make excellent switches for this purpose. The only issue of potential concern is their nonzero on-resistance, as per Eq. (1).

The DAC of Fig. 13 uses a CMOS binary counter both to provide the SPDT switching action and to increment the input word  $b_1 b_2 b_3 \dots b_8$  in counter fashion. Using Eq. (7), you can convince yourself that  $v_O$  will then be a staircase approximation to a saw-tooth waveform. This waveform is readily observed with the oscilloscope.

To gain better insight into circuit operation, we shall start out with the bottom terminals of all the  $2R$  resistors (nodes labeled  $b_1$  through  $b_8$ ) initially connected to ground, and we shall subsequently lift one leg at a time off ground and connect it to the corresponding counter output  $Q_i$ , starting with the MSB leg ( $b_1$ ) and proceeding one leg at a time toward the less significant legs. Note that we are powering our circuit with a 10-V supply ( $V_{DD} = V_{REF} = 10\text{ V}$ ), indicating that the pulse generator must be adjusted accordingly, with  $v_{CK}$  alternating between  $0\text{ V}$  and  $10\text{ V}$ . The purpose of the  $10\text{-k}\Omega$  series resistor is to protect the counter's clock input against possible overdrive. Finally, note that the output  $v_O$  is taken from the upper left corner of the ladder.

**M20:** With power off, assemble the circuit of Fig. 13 (use  $R = 10\text{ k}\Omega$  and  $2R = 20\text{ k}\Omega$ ), but with the bottom terminals of all the  $2R$  resistors (nodes labeled  $b_1$  through  $b_8$ ) initially connected to ground. As usual, keep leads short, and bypass the  $10\text{-V}$  supply to ground via a  $0.1\text{-}\mu\text{F}$  capacitor. Also, adjust the pulse generator for a  $50\text{-kHz}$  clock alternating between  $0\text{ V}$  and  $10\text{ V}$ . Then, apply power, and while monitoring  $Q_{11}$  (Pin 15) with Ch. 1 of the oscilloscope and  $v_O$  with Ch. 2, trigger from Ch. 1, proceed as follows:

- Lift  $b_1$  off ground and connect it to  $Q_{11}$  (Pin 15). You will observe that  $v_O$  is a 2-level staircase (in fact, it is a square wave) with the same frequency as  $Q_{11}$ , and amplitude  $(1/2)V_{REF}$ , or  $5\text{ V}$ .
- Lift  $b_2$  off ground and connect it to  $Q_{10}$  (Pin 14). This will add to the existing output a square wave with twice the frequency of  $Q_{11}$  and amplitude  $(1/4)V_{REF}$ , turning  $v_O$  into a 4-level staircase with step size  $(1/4)V_{REF}$ , or  $2.5\text{ V}$ .

- Lift  $b_3$  off ground and connect it to  $Q_9$  (Pin 12). This will add to the existing output a square wave with *four times* the frequency of  $Q_{11}$  and amplitude  $(1/8)V_{REF}$ , turning  $v_O$  into an 8-level staircase with step size  $(1/8)V_{REF}$ , or 1.25 V.
- Lift  $b_4$  off ground and connect it to  $Q_8$  (Pin 13). This will add to the existing output a square wave with *eight times* the frequency of  $Q_{11}$  and amplitude  $(1/16)V_{REF}$ , turning  $v_O$  into a 16-level staircase with step size  $(1/16)V_{REF}$ , or 0.625 V.
- Continue in the same manner with  $b_5, b_6, \dots$ . However, as you keep adding bits, you will notice that the step sizes become less and less uniform, until at least one step will differ from its anticipated value by at least half such a value. The most critical step is the one in the middle of the staircase, corresponding to the transition from  $b_1b_2b_3b_4 = 011\dots1$  to  $b_1b_2b_3b_4 = 100\dots0$ , also called the *major carry transition*. Clearly, when the steps become excessively non-uniform, it makes no sense adding more bits. At what bit ( $b_5$ ?  $b_6$ ?  $b_7$ ?  $b_8$ ?) does the major-carry step size of your DAC differ from its nominal value by at least  $\pm 50\%$ ?
- What happens to the output waveform if you sweep the clock up and down the frequency spectrum? In particular, do you envision a practical upper limit to the useful frequency of operation? Based on what criteria? Likewise, is there any lower limit?

**C21:** Exploiting unique properties of the  $R$ - $2R$  ladder, explain why connecting a given bit in the above sequence adds to the output a square wave with *twice* the frequency and *half* the amplitude of that added by the previous bit in the sequence. What are the reasons for the progressively uneven step sizes? What are the reasons for the deterioration in step sizes and shapes as the clock frequency is increased beyond a certain limit?