Homework 2

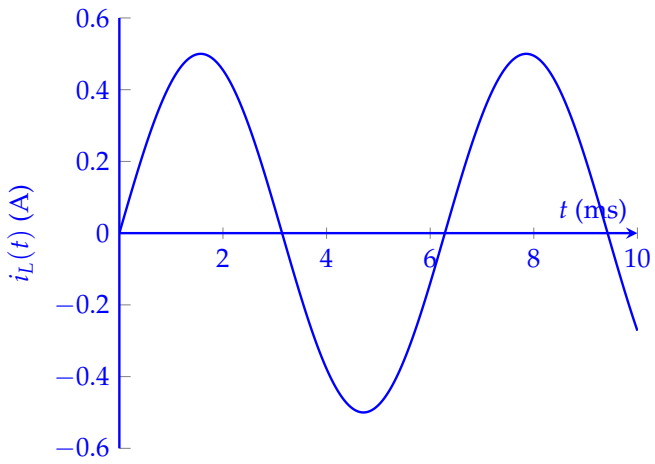
This homework is due on Saturday, September 9, 2023 at 11:59PM. Self-grades and HW Resubmissions are due the following Saturday, September 16, 2023 at 11:59PM.

The homework relies on material covered in lecture on inductors (**08/31**) and transistors (**09/05**) as well as the corresponding notes, **Note 3** and **Note 4** respectively.

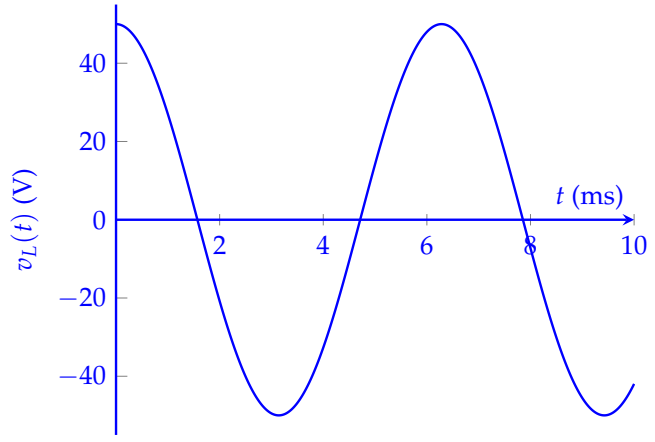
1. **Hambley P3.49**

   The current in a 100 mH inductance is given by $0.5\sin(1000t)$A. Find expressions and sketch the waveforms to scale for the voltage, power, and stored energy, allowing t to range from 0 to 3π ms. The argument of the sine function is in radians.

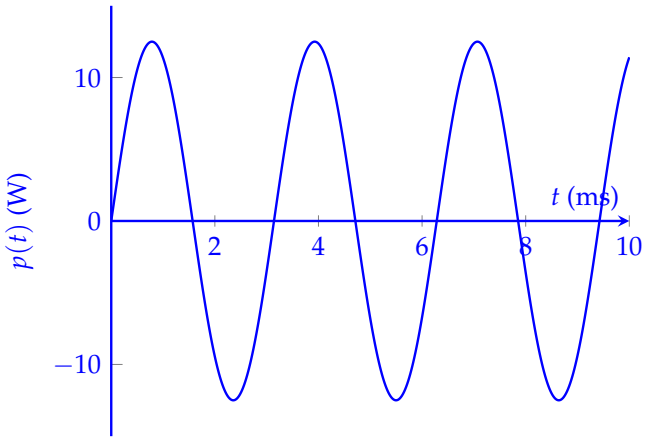
   Solution: Below is the plot for $i_L(t)$:



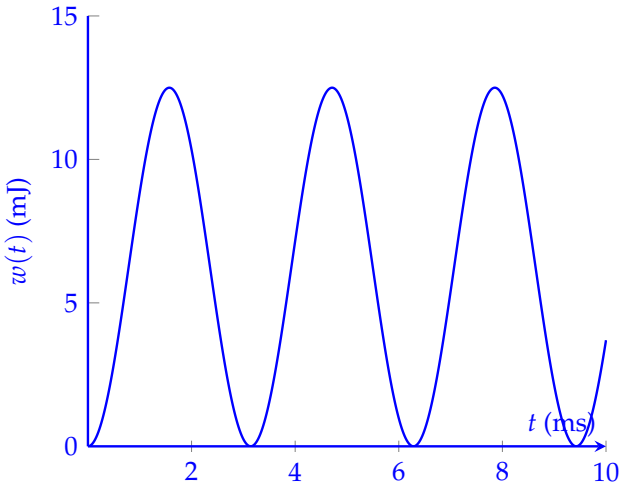
   Using this and the fact that $v_L(t) = L\frac{di_L(t)}{dt} = 50\cos(1000t)$, we have the following plot for voltage:



   Next, we have $p(t) = v_L(t)i_L(t) = 25\cos(1000t)\sin(1000t) = 12.5\sin(2000t)$, which is plotted below:



Lastly, we have $w(t) = \frac{1}{2}L(i_L(t))^2 = 0.0125\sin^2(1000t)$, which is plotted below:



2. Hambley P3.55

What value of inductance corresponds to an open circuit, assuming zero initial current? Explain your answer. Repeat for a short circuit.

Solution: In an open circuit, $i = 0$, we have that

$$i = \frac{1}{L}\int_{t_0}^{t} v(t')\,\mathrm{d}t' + i(t_0) \tag{1}$$

and $i(t_0) = 0$. Hence, it must be the case that $L \to \infty$.
In a short circuit, $v = 0$, so we have

$$v = L\frac{\mathrm{d}i}{\mathrm{d}t} = 0 \tag{2}$$

so $L = 0$.

3. Hambley P4.21

Solve for the steady-state values of i_1, i_2, and i_3 in the circuit shown in Figure 1.

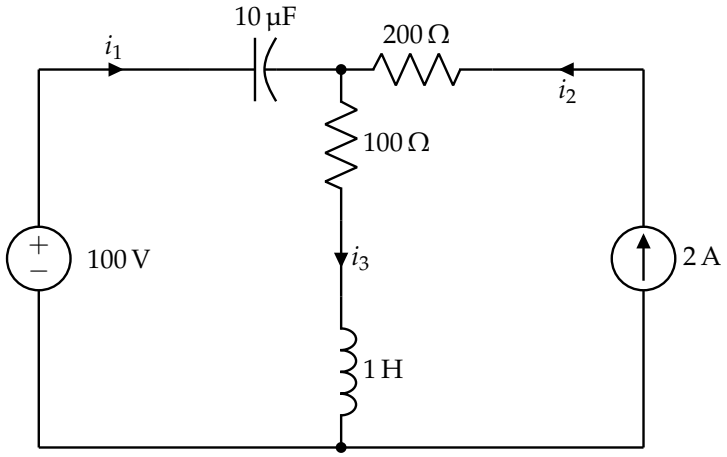
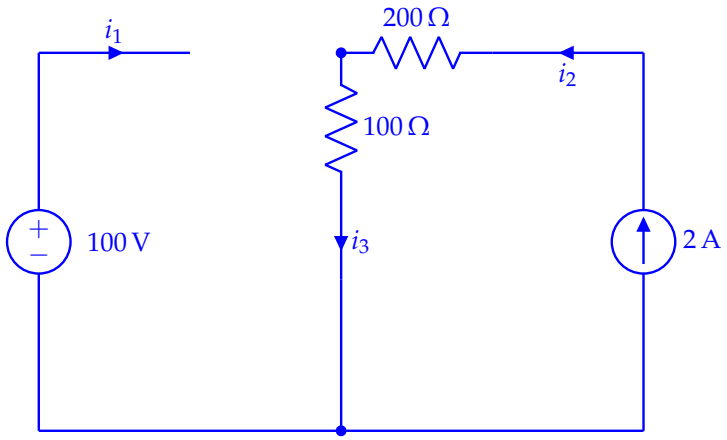


Figure 1: P4.21

Solution: At steady state, the equivalent circuit is



so $i_1 = 0$ and $i_2 = i_3 = 2$ A.

4. Hambley P4.41

Determine expressions for and sketch $v_R(t)$ to scale versus time for the circuit of Figure 2. The circuit is operating in steady state with the switch closed prior to $t = 0$. Consider the time interval $0 \leq t \leq 1$ ms.

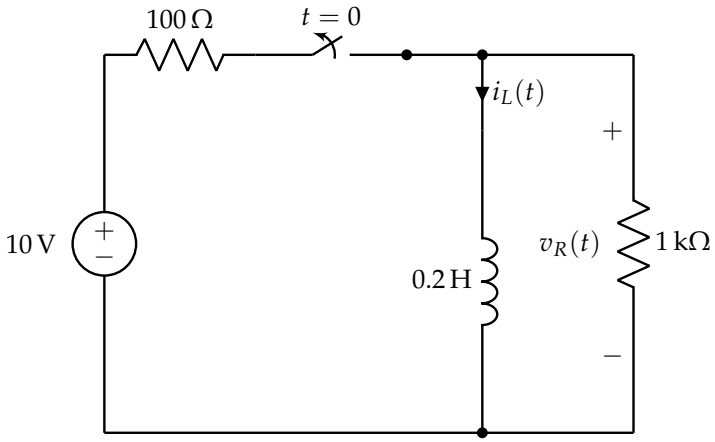
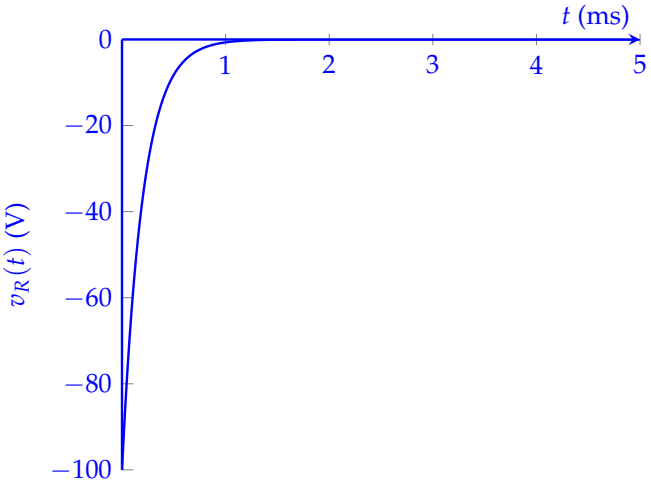


Figure 2: P4.41

Solution: In the steady state the inductor acts as a short circuit (essentially shorting the 1 kΩ resistance on the right). Then $i_L = \frac{10}{100} = 0.1$ A.

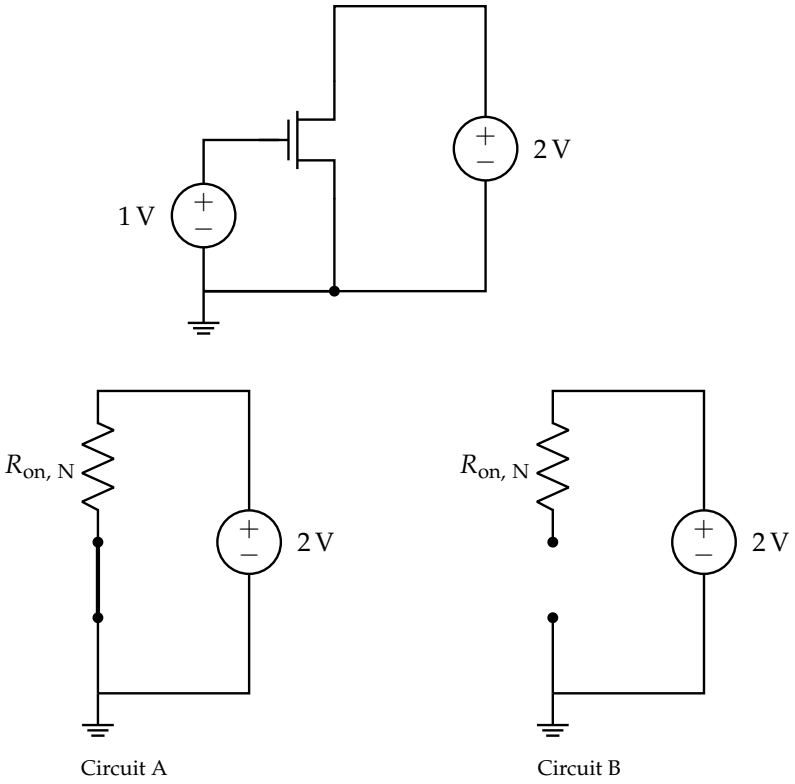
As the switch is opened, the circuit essentially becomes a R-L circuit with an initial voltage of $v_R(0) = -1000 \times 0.1 = -100$ V. Then from $L\frac{di}{dt} + iR = 0$, we get $i = i(0)e^{-\frac{R}{L}t}$. Thus, $v_R(t) = -100e^{-5000t}$. A plot of this is shown below:



5. Transistor Behavior

For all NMOS devices in this problem, $V_{tn} = 0.5\,V$. For all PMOS devices in this problem, $|V_{tp}| = 0.6\,V$.

(a) Which is the equivalent circuit as seen from the voltage source on the right-hand side of the circuit? **Fill in the correct bubble.**

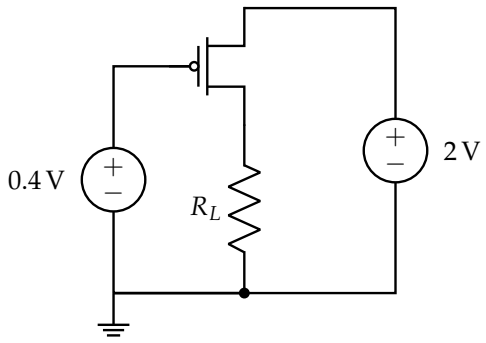


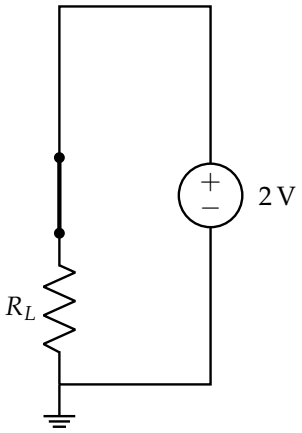
Circuit A Circuit B

	A	B
Equivalent Circuit	◯	◯

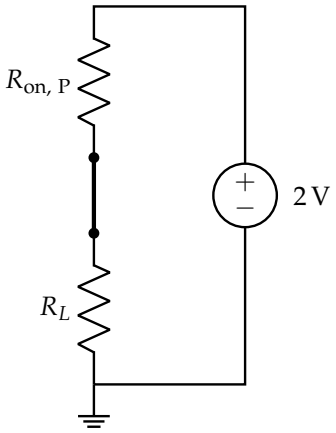
Solution: For the NMOS, $V_{GS} = 1\,V > V_{tn} = 0.5\,V$, so the NMOS transistor is on.
Thus circuit A is equivalent.

(b) Which is the equivalent circuit as seen form the voltage source on the right-hand side of the circuit? **Fill in the correct bubble.**

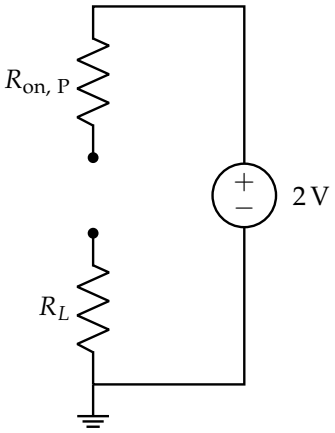




Circuit A



Circuit B

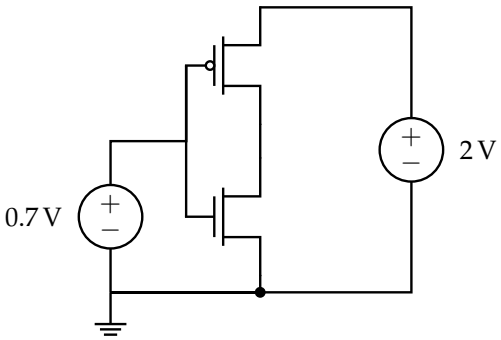


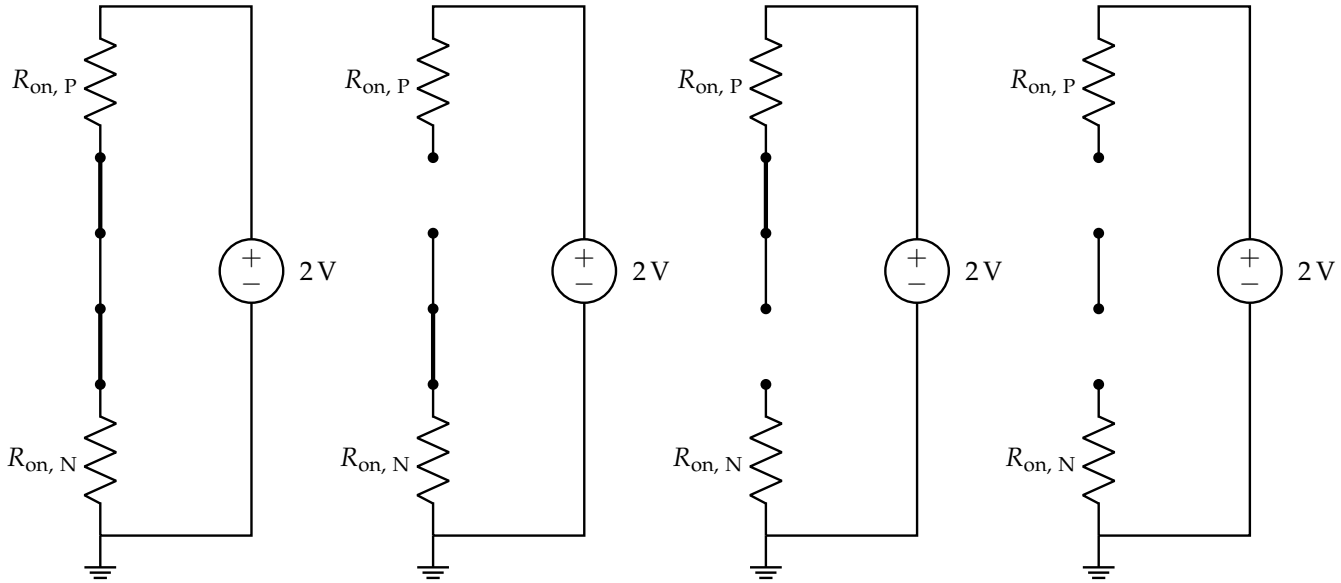
Circuit C

	A	B	C
Equivalent Circuit	◯	◯	◯

Solution: For the PMOS transistor, $|V_{GS}| = 1.6\,\text{V} > |V_{tp}| = 0.6\,\text{V}$, so the PMOS transistor is on. Thus circuit B is equivalent.

(c) Which is the equivalent circuit as seen form the voltage source on the right-hand side of the circuit? **Fill in the correct bubble.**





	A	B	C	D
Equivalent Circuit	◯	◯	◯	◯

Solution: For the PMOS transistor, $|V_{GS}| = 1.3\,\text{V} > |V_{tp}| = 0.6\,\text{V}$, so the PMOS transistor is on. For the NMOS transistor, $V_{GS} = 0.7\,\text{V} > V_{tn} = 0.5\,\text{V}$, so the NMOS transistor is on.

Note that in this case, both transistors are on.

Thus circuit A is equivalent.

Aside: In digital logic, it is usually undesirable to have this state in your system for several reasons. First, the output voltage of the inverter (the voltage at the shared drain of the NMOS and PMOS) will not be either 0 or V_{DD}, which means the output voltage is not at 'true' binary value. In addition, we now have a direct current path through the NMOS and PMOS transistors from VDD to ground. This will burn a lot of power! In reality, all inverters briefly transition through this state where both NMOS and PMOS are on when the inputs change from 1 to 0 or 0 to 1.

6. Successive Approximation Register Analog-to-Digital Converter (SAR ADC)

An analog-to-digital converter (ADC) is a circuit for converting an analog voltage into an approximate digital representation of that voltage. One commonly used circuit architecture for analog-to-digital converters is the Successive Approximation Register ADC (SAR ADC), which you will see in Lab 3. An N-bit SAR ADC converts an input analog voltage to an N-bit binary string between 0 and $2^N - 1$. This binary string represents an integer, which in term approximates the value of our analog input voltage.

The SAR ADC does this by following one of the key themes in 16B: reducing a problem into subproblems that we already know how to solve. In this case, the two ingredients are the DAC (digital to analog converter) that we saw in HW 1, and a binary search tree that you saw in 61A. As you remember from 61A, the key operation required for a binary search is dividing a group into two halves, solving the problem at the current step with a less-than/greater-than comparison, and descending into one of the halves. We continue this until we can no longer subdivide the problem. The comparison operation is therefore key to a binary search tree. Fortunately, we have a circuit element from 16A that lets us do that: a comparator.

Explicitly, the SAR ADC implements the binary search algorithm by feeding trial digital codes into a **DAC**, like the one we analyzed in Homework 1. The circuit then takes the resuling analog voltage from the DAC and compares it with the analog input voltage using a **comparator**. It then uses feedback (**SAR logic**) to adjust the DAC output voltage to get as close as possible to the input analog voltage, step by step. The algorithm starts by determining the most significant bit (MSB), which is the bit with the largest binary weight (i.e. furthest to the left in a traditional binary number), and then moving on to determine the next bit.

If this is not clear to you, think about how you would play 20 questions to guess an integer between 0 and $2^{20} - 1$, which is approximately 1 million. This is a game where you have to guess what number your friend is thinking of, and they can only tell you if your guess is too high or too low. You have 20 guesses to get your number as close as possible to your friend's. Would you start by guessing 0, then 1, then 2, etc.? Or would you start in the middle, let's say 500,000, see if you're too high or low, then move into the next half, e.g. 250,000 or 750,000 depending on your result? The latter approach of divide-and-conquer is a faster way of solving the problem! The SAR ADC is basically a circuit implementation of this game. The input voltage is the "number" your friend has in mind, the DAC code is your guess, the comparator is your friend's response (too high vs. too low), and the SAR logic used to adjust the next code is you deciding what to guess next. For a 20-bit SAR DAC, you have 20 guesses; for an N-bit SAR DAC, you have N guesses.

Figure 9 illustrates a high-level circuit diagram of a SAR ADC. The analog input voltage is one of the inputs to the comparator. The other comparator input, V_{DAC}, is the DAC voltage output, i.e. the analog representation of your code/guess. The comparator compares these two values and outputs a logical high (1) if $V_{IN} \geq V_{DAC}$ or a logical low (0) if $V_{IN} < V_{DAC}$. This comparator decision then feeds into the SAR logic, implemented in a microcontroller in this case. This logic is basically the brain that implements the binary search pattern, deciding which bits in the code need to be changed. The updated code then propagates back to the DAC, which updates V_{DAC}, and the cycle continues N times. Once this is done, the final N-bit output code comes out of the register storage units for use by other circuits. If you're wondering what V_{REF} does for the DAC, recall from Homework 1 that it tells

the DAC what the maximum possible input voltage is.

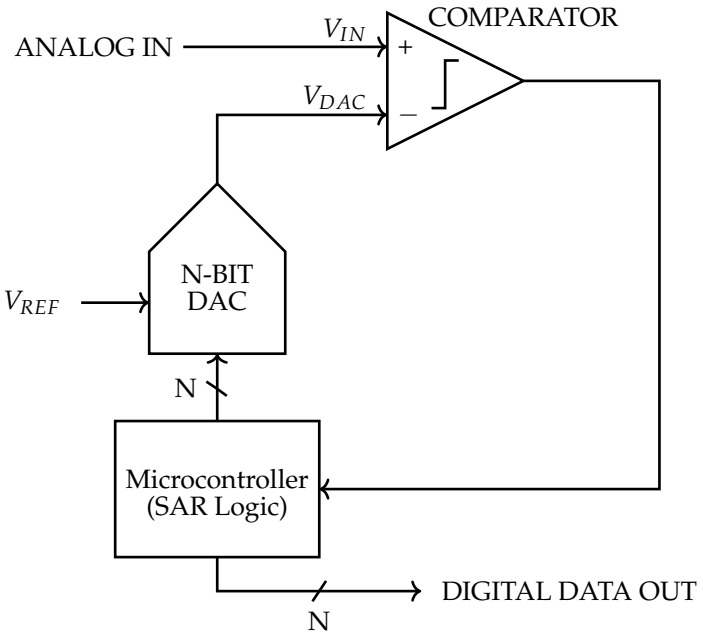


Figure 9: SAR ADC circuit diagram

Here is an illustration of the algorithm, which the SAR logic takes care of.

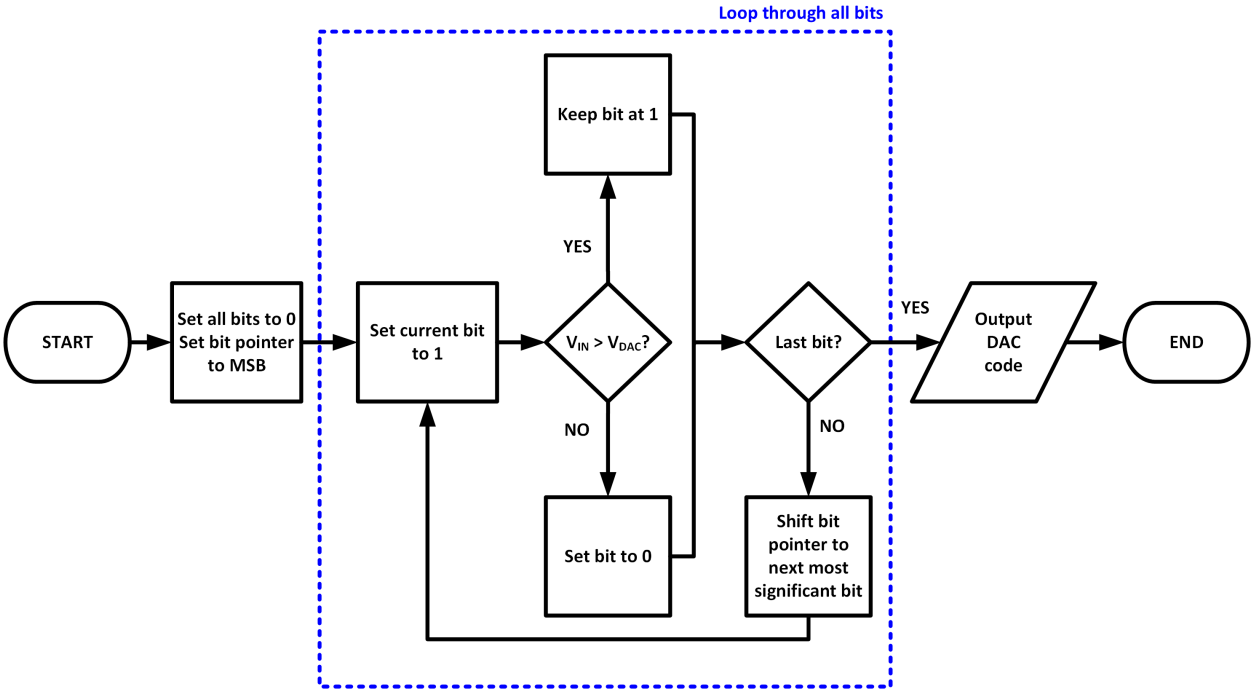


Figure 10: Flow chart of SAR ADC algorithm.

Let's look at a 3-bit SAR ADC as an example. The voltage transfer curve of the 3-bit SAR ADC is

shown in Figure 11. For $V_{\text{IN}} = 0.3V_{\text{REF}}$, the operation of SAR ADC is shown in Figure 12.

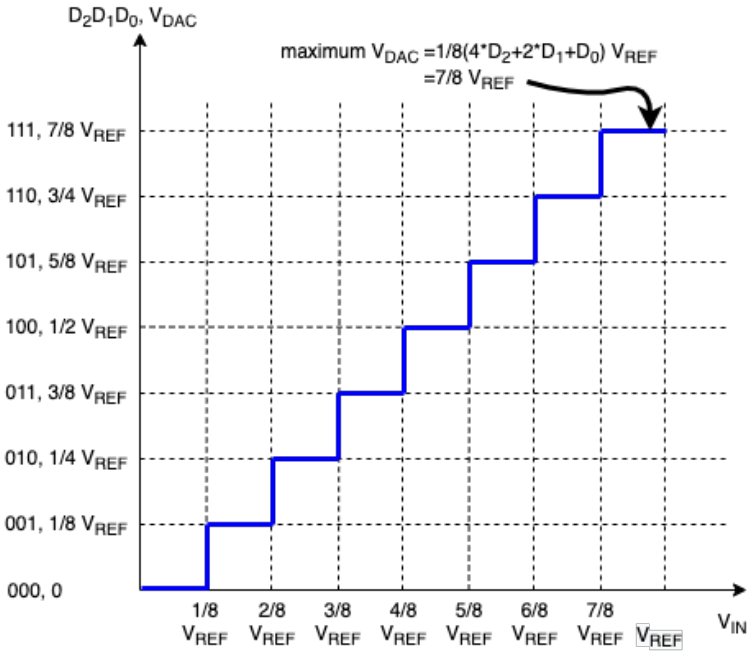


Figure 11: Voltage transfer curve of 3-bit SAR ADC: $D_2D_1D_0$ is the output digital code of the ADC. Notice that the maximum voltage of the V_{DAC} is $\frac{2^N-1}{2^N}V_{\text{REF}}$ where N is the number of bits ($N = 3$ in this case)

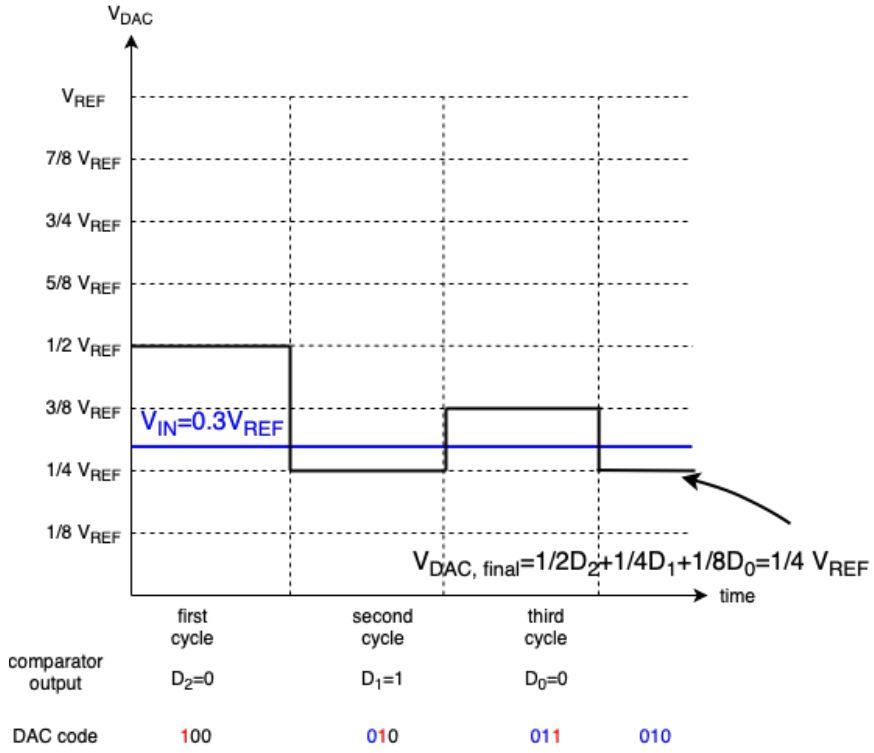


Figure 12: Timing diagram of SAR output code. Red digit in DAC code is the bit being decided in the cycle; blue digits are the bits determined by the previous conversion cycles.

From the timing diagram in Figure 12, it can be seen that the final digital output code is 010 which represents the closest DAC output voltage $\left(\frac{1}{4}V_{\text{REF}}\right)$ to the input analog voltage $V_{\text{IN}} = 0.3V_{\text{REF}}$ with a 3 bit binary representation.

We will now analyze the operation of a 4-bit SAR ADC with the input voltage range between 0 V and 5 V and output code range between 0000 (0) and 1111 (15). In other words, V_{REF} is 5 V and $0\,\text{V} < V_{\text{in}} < 5\,\text{V}$ in Figure 9.

(a) **If the ADC output code is 0000, what is the corresponding DAC voltage? What about code 1111? Code 1001?** *(HINT: Try to draw the transfer curve for 4-bit SAR ADC, similar to that for the 3-bit SAR ADC in Figure 11.)*

Solution: Output code 0000 corresponds to $V_{\text{DAC}} = 0\,\text{V}$.
Output code 1111 corresponds to $V_{\text{DAC}} = \frac{1\times2^3+1\times2^2+1\times2^1+1\times2^0}{2^4} \times V_{REF} = \frac{15}{16} \times 5\,\text{V} = 4.6875\,\text{V}$.
Output code 1001 corresponds to $V_{\text{DAC}} = \frac{1\times2^3+0\times2^2+0\times2^1+1\times2^0}{2^4} \times V_{REF} = \frac{9}{16} \times 5\,\text{V} = 2.8125\,\text{V}$.

(b) **If the input analog voltage is $V_{\text{IN}} = 3\,\text{V}$, what is the ratio between the input voltage V_{IN} and the maximum input voltage $V_{\text{REF}} = 5\,\text{V}$? What should be the output code of the SAR ADC?**

Solution: The ratio is $\frac{V_{\text{IN}}}{V_{\text{REF}}} = \frac{3}{5} = 0.6$.
The output code can be computed in 4 cycles using the SAR ADC algorithm.
In the first cycle, the DAC code is 1000, resulting in comparator output $D_3 = 1$ since $0.6 > \frac{8}{16}$.
In the second cycle, the DAC code is 1100, resulting in comparator output $D_2 = 0$ since $0.6 < \frac{12}{16}$.
In the third cycle, the DAC code is 1010, resulting in comparator output $D_1 = 0$ since $0.6 < \frac{10}{16}$.
In the fourth cycle, the DAC code is 1001, resulting in comparator output $D_0 = 1$ since $0.6 > \frac{9}{16}$.
Hence the final DAC output code is $D_3D_2D_1D_0 = 1001$.

(c) Again, if the input analog voltage is $V_{\text{IN}} = 3\,\text{V}$, draw the operation of a 4-bit SAR ADC resolving its output code (see Figure 12 as reference). Specifically:

i. **Plot the output voltage of the DAC in the timing diagram in Figure 13.**
ii. **Fill out the output of the comparator in each conversion cycle.**
iii. **Fill out the ADC output code ($D_3D_2D_1D_0$) in each conversion cycle.**

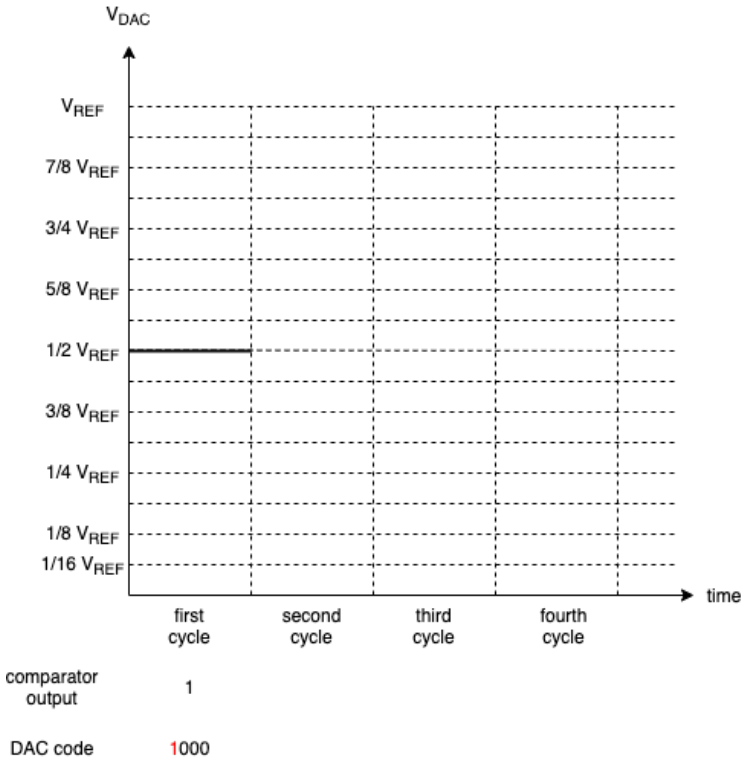


Figure 13: SAR ADC V_{DAC} timing diagram

Solution: See Figure 14.

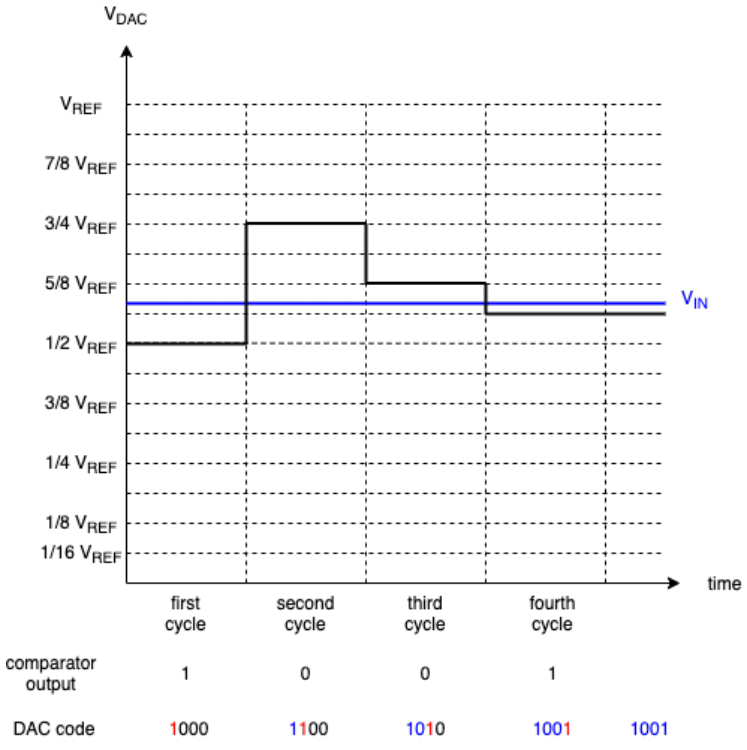


Figure 14: SAR ADC V_{DAC} timing diagram answer

Contributors:

- Sidney Buchbinder.
- Anant Sahai.
- Yi-Hsuan Shih.
- Vladimir Stojanovic.
- Ayan Biswas.
- Wahid Rahman.