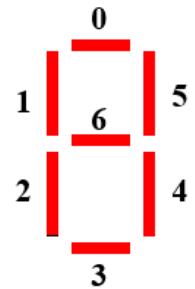
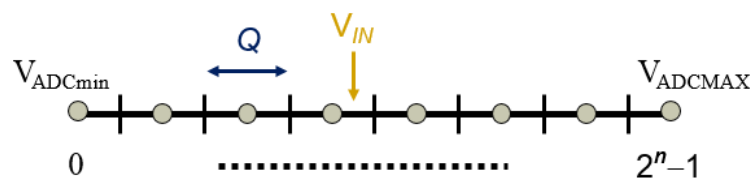
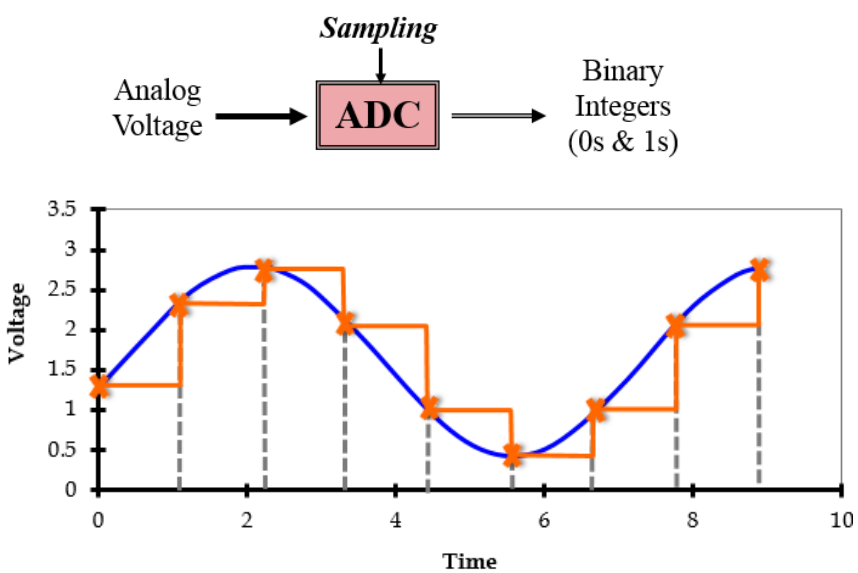
FROM LAST TIME…

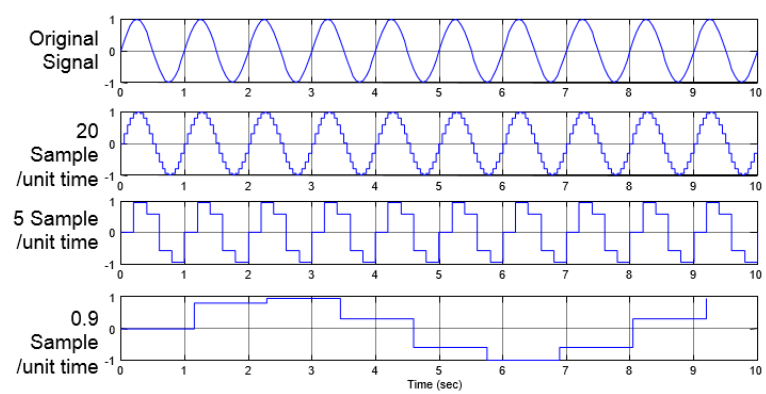
System Interfacing (A/D and D/A Conversion)

- Why do A/D and D/A matter?
- Data representation
- Number systems
- Analog ⇔ Digital conversion
- Sampling and Aliasing





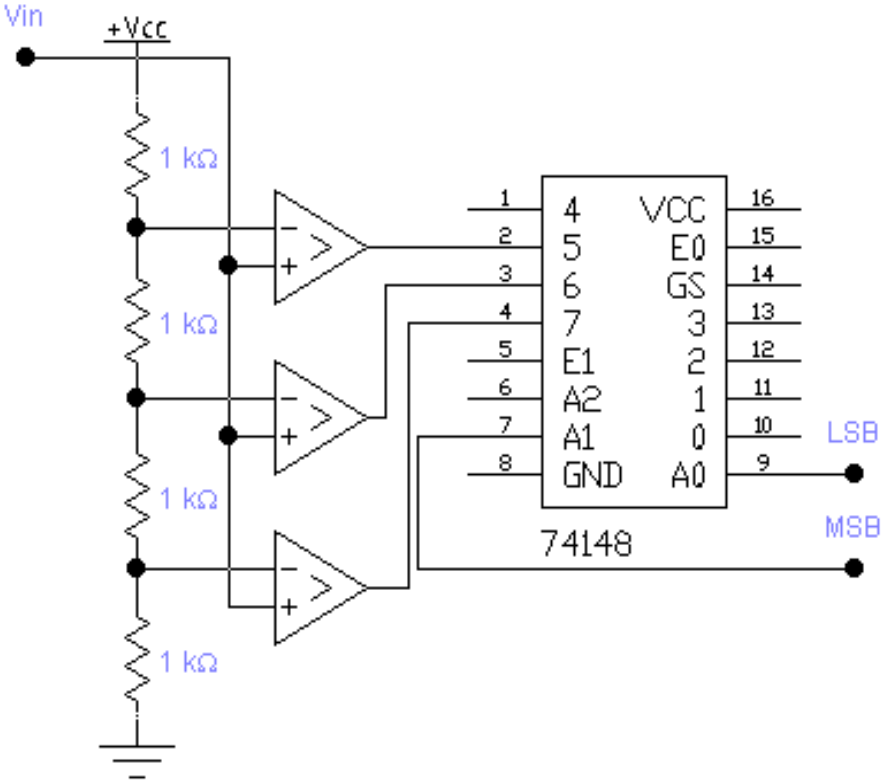




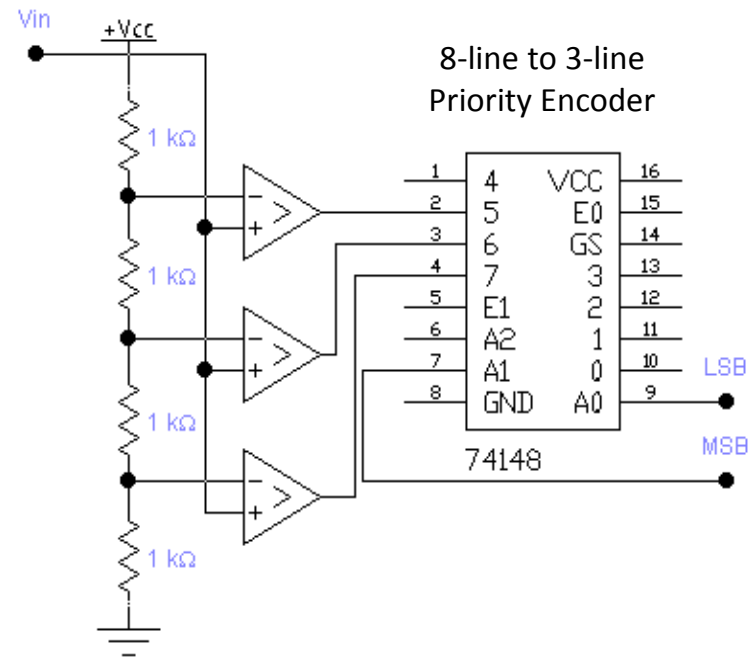
ANALOG-TO-DIGITAL (A/D) CONVERSION

Flash ADC

- Uses comparators to determine input voltage range.
- Gate logic converts comparator outputs to digital value.
- Fast! Typical conversion time: 10 – 500 nsec.
- Typically, 4 to 8 bit precision (8 bits requires 254 comparators).



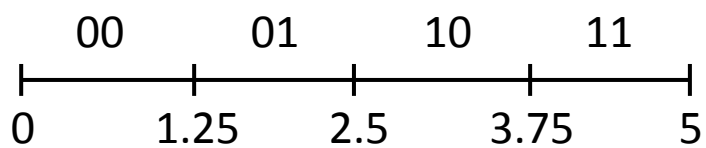
ANALOG-TO-DIGITAL (A/D) CONVERSION



Flash ADC

- Example: *2 bit Flash ADC*

  - Code:



  - Needs: 3 comparators

    C1: V > 1.25
    C2: V > 2.5
    C3: V > 3.75

Use truth table to get output values:

C3	C2	C1	MSB	LSB
0	0	0	0	0
0	0	1	0	1
0	1	0	X	X
0	1	1	1	0
1	0	0	X	X
1	0	1	X	X
1	1	0	X	X
1	1	1	1	1

ANALOG-TO-DIGITAL (A/D) CONVERSION

Flash ADC

- Need to provide sample-and-hold at the input -- if input is changing during conversion, erroneous values will be produced.
- Output valid for only a short time after input is held.
- Requires $2^N - 1$ comparators.
- Adjacent comparators must have monotonic range change.

ANALOG-TO-DIGITAL (A/D) CONVERSION

Successive Approximation ADC:

- Workhorse method.
- Used for wide variety of applications
- Typical conversion time: 1 – 100 μsec. Slower than flash ADC.
- Easily extensible to higher precision.
- Precision is limited by the quality of the components.

SUCCESSIVE APPROXIMATION ADC

Basic idea: Check bits starting from the high order bit (MSB).

- Algorithm:

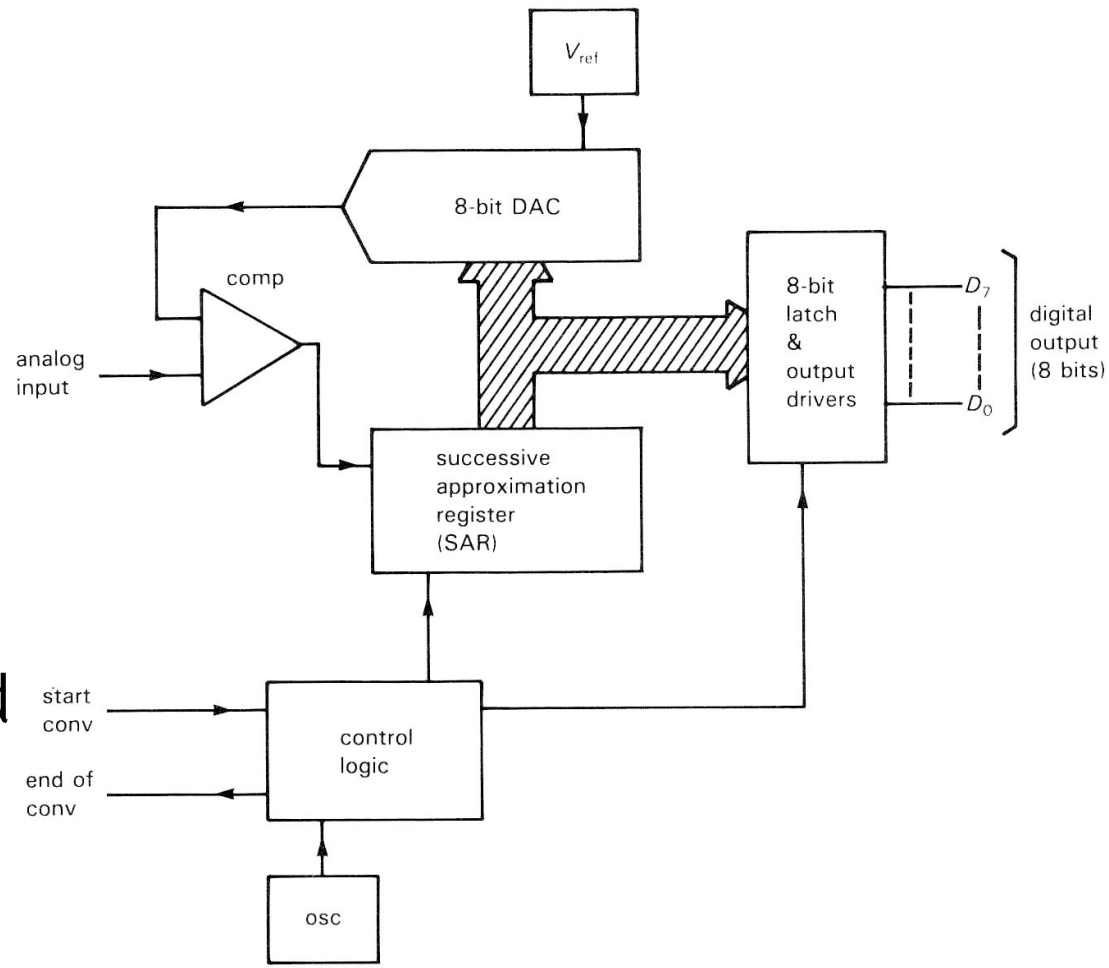
```
START CONVERSION
   SET Result to 0
   FOR i = N-1 TO 0
      SET i-th bit of Result to 1
      IF INPUT VOLTAGE < DtoA(Result)
         SET i-th bit of Result to 0
   END FOR-loop
   OUTPUT Result
END CONVERSION
```

This is a form of *interval halving*.

SUCCESSIVE APPROXIMATION ADC

General structure:

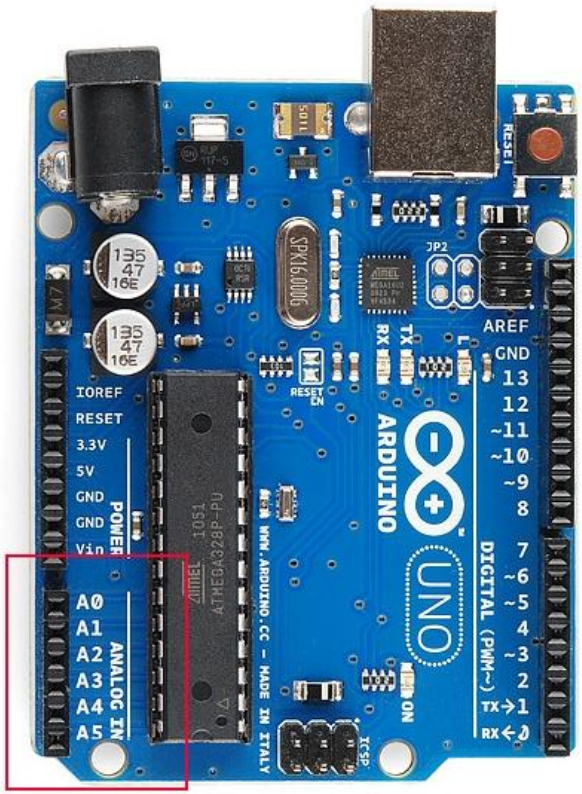
- Successive-approximation converters are quite expensive.
- Usually used with a multiplexer -- many channels feed to a single converter.
- Effective conversion speed for multiplexed ADC depends on number of channels used.
- Sample-and-hold normally precedes the converter.

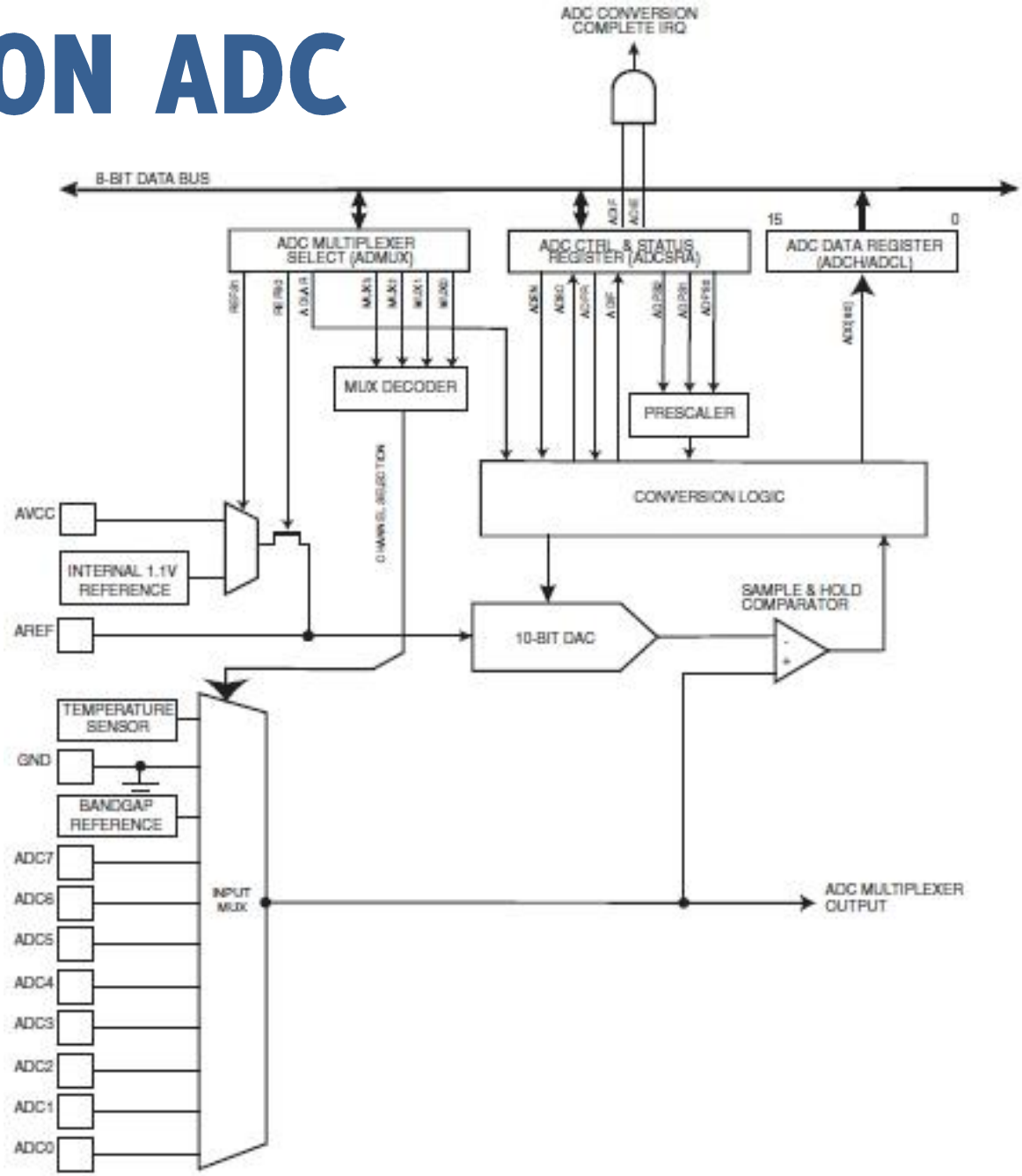


SUCCESSIVE APPROXIMATION ADC

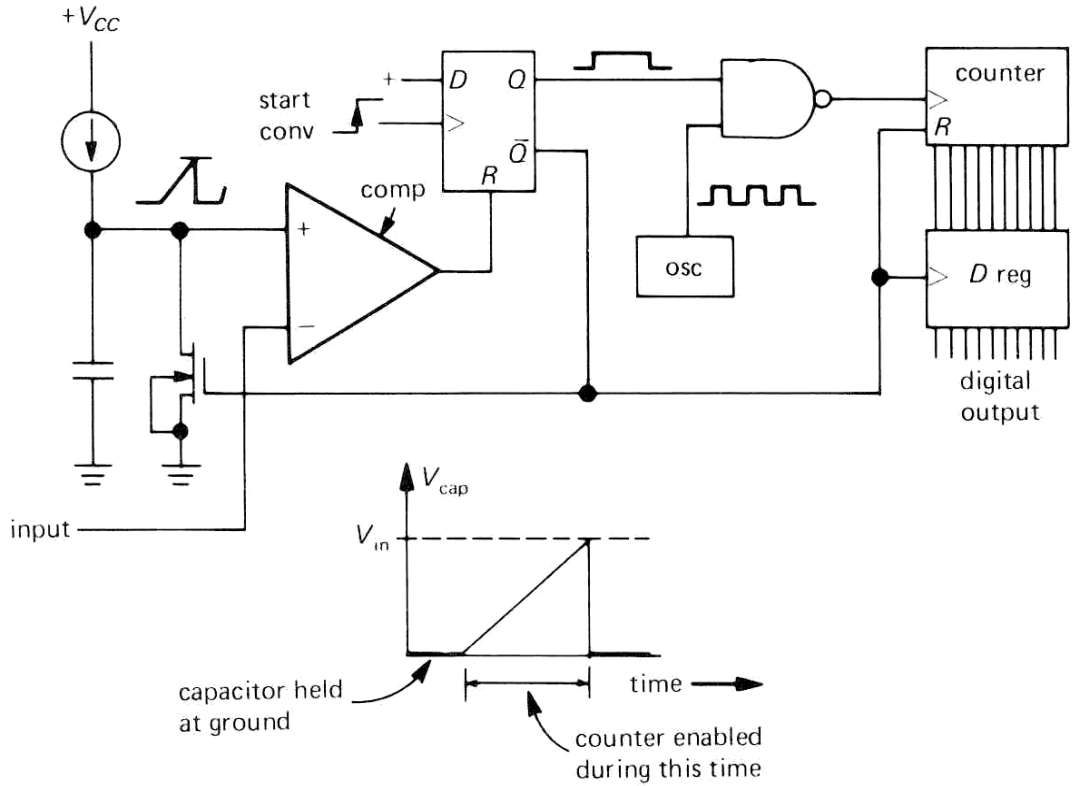
Arduino ADC

6-channel 10-bit ADC





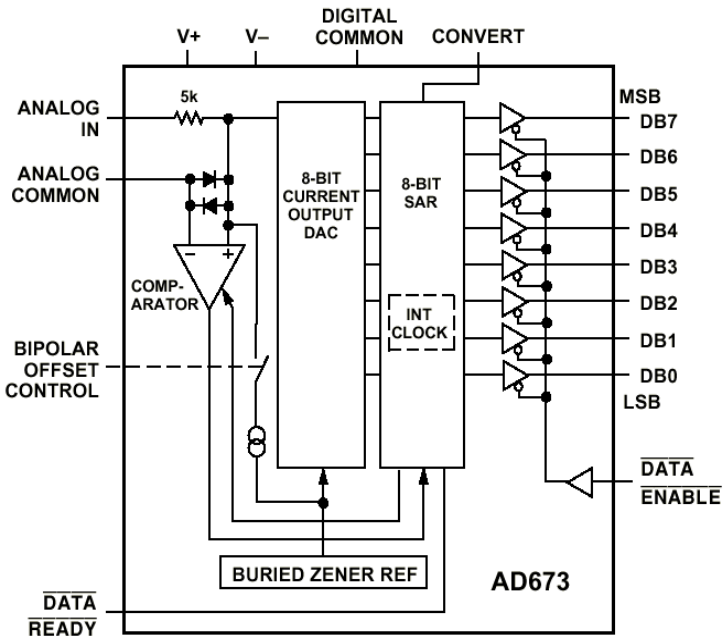
INTEGRATING CONVERTERS

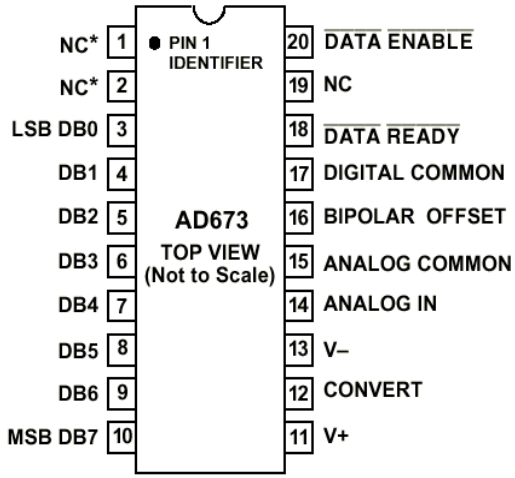


- Slowest of commonly used A/D converters. Typical conversion time is many milliseconds.
- Can be made very accurate and precise – used in DVMs (several conversions per second).
- Uses timing to determine digital value of unknown (input) voltage.

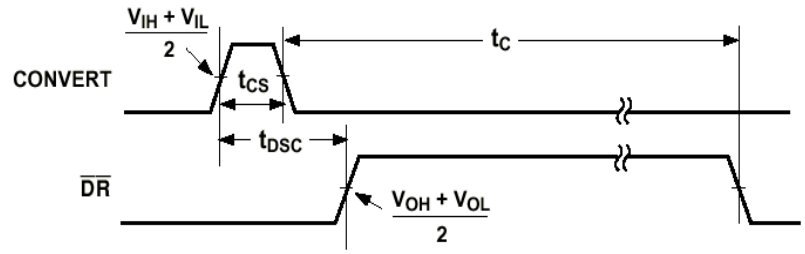
INTERFACING WITH AN ADC

- Successive approximation ADC – use AD673 as example

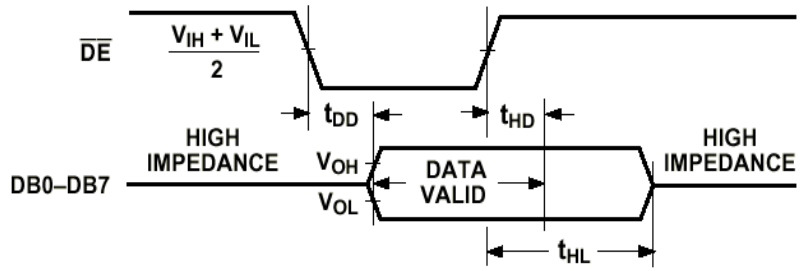


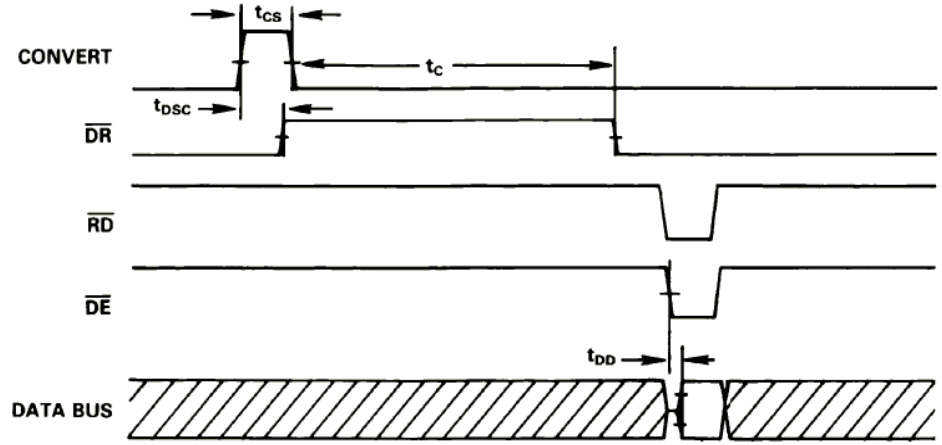


Convert timing

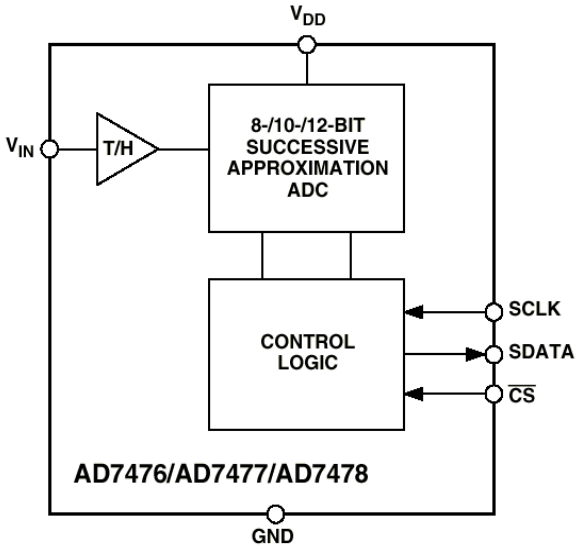


Reading timing



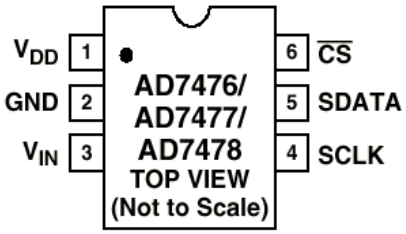


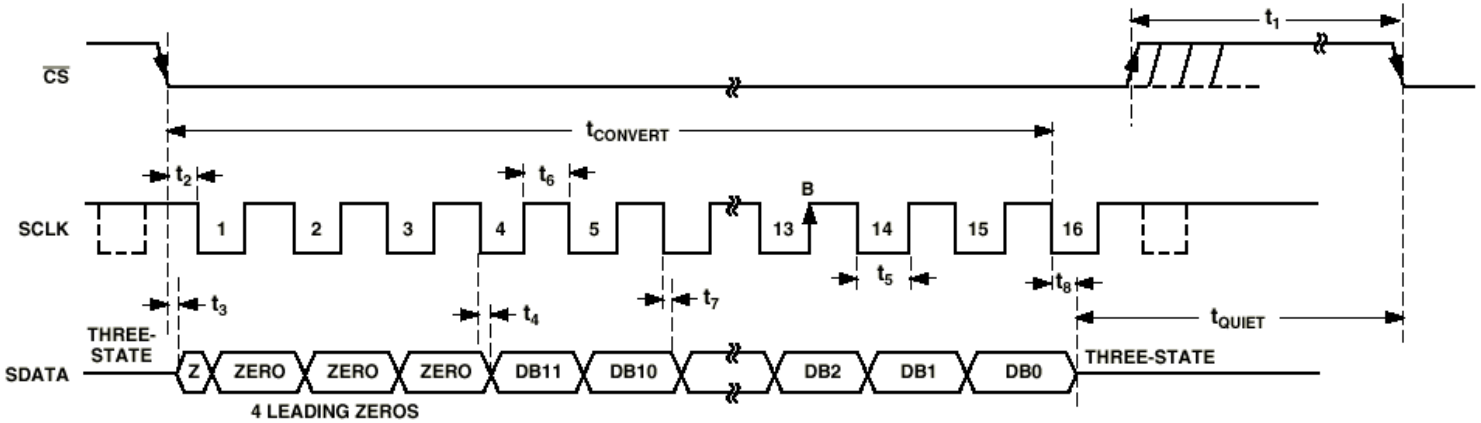
ANALOG-TO-DIGITAL (A/D) CONVERSION



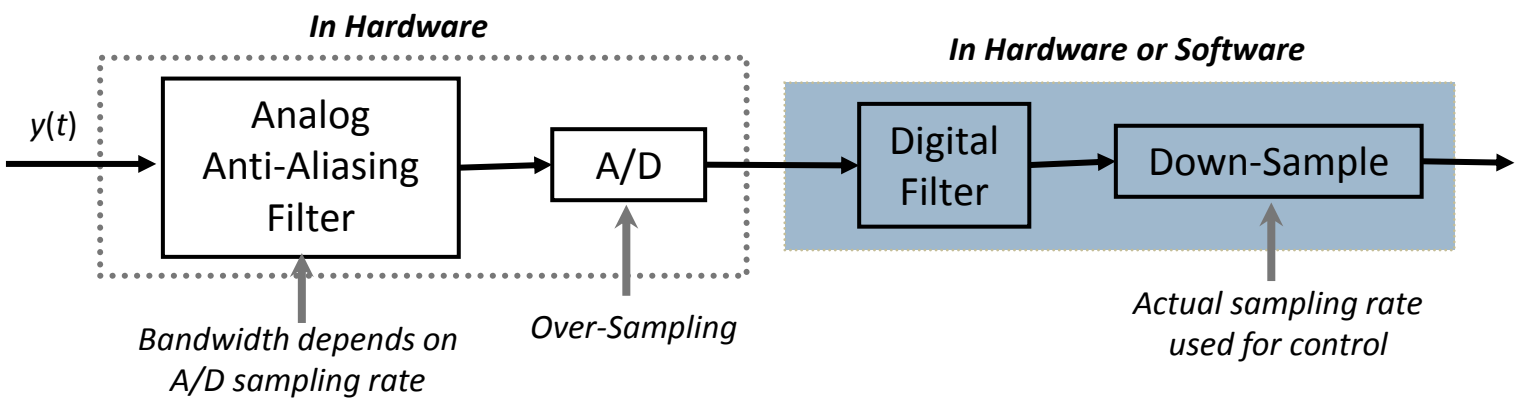
ADC with Serial Output

- Reduce pinouts and package size
- Can be easily interfaced with a microcontroller or a microprocessor with built in serial interface (SCLK, SDATA, T/R)
- Sample rate limited by the maximum SCLK rate.
- Use AD7476 as example – 1 MSPS 6 pin ADC





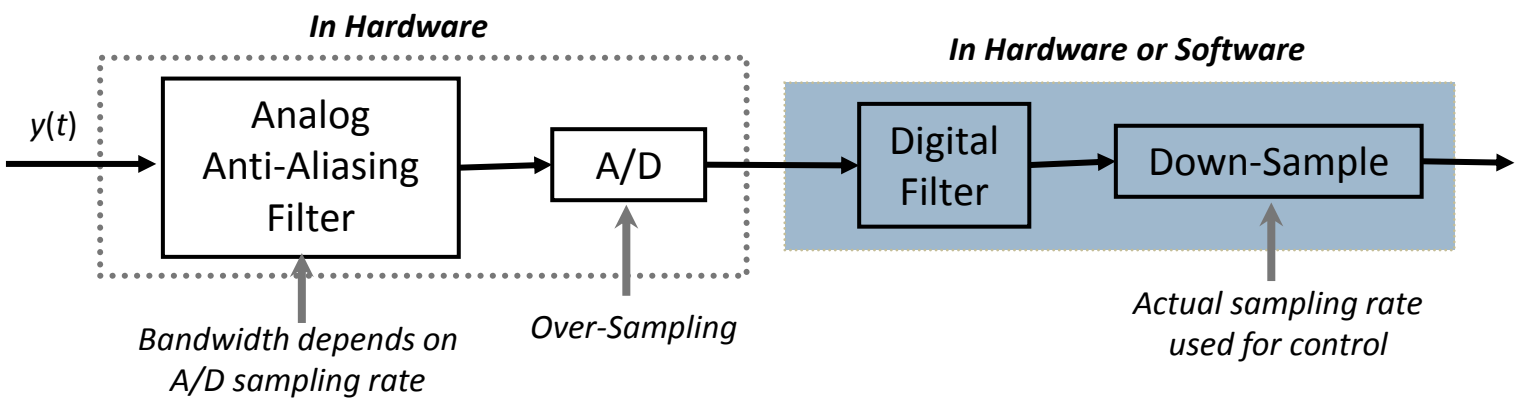
OVER-SAMPLING AND SUB-SAMPLING



If A/D conversion is fast enough, it is possible to sample much faster in the converter than is needed by the application

- Digital filter can then be applied to remove unwanted spectral components.
- Anti-aliasing filter can be split between analog and digital filtering
- DSP is often used to implement digital filter.

OVER-SAMPLING AND SUB-SAMPLING



Over-sampling converters may be less expensive or flexible, but they do not have better performance than other converters.

- Stream of over-sampled data has more data values than can be used.
- Sub-sampling picks out every Nth data value and throws away $(N-1)/N$ of the data (N is the over-sampling ratio).

OVERSAMPLING

- One benefit of oversampling is that it spreads out *broadband* noise over a wide frequency range, thus lowering the "noise floor." Other sources of noise and other errors cannot be removed by oversampling. Thus, sampling speed can be traded for amplitude resolution.

- Oversampling converters do not have any better performance than other D/A converters. They use knowledge of noise characteristics to get 24-bit performance, say, out of a 20-bit converter, by sampling much faster than the target sampling rate.

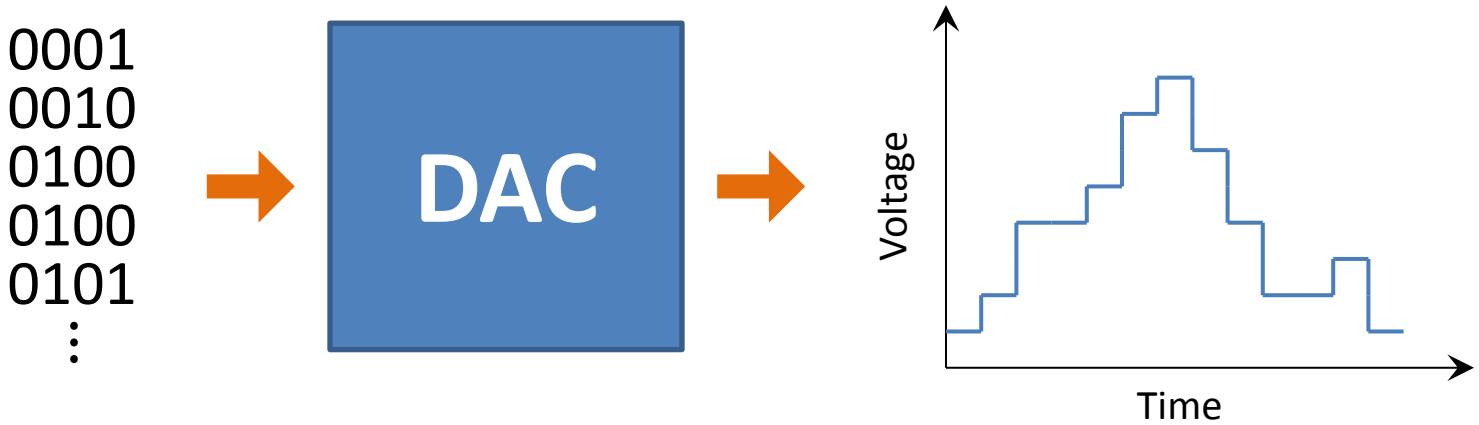
UNIT 5:
SYSTEM INTERFACING

PART B:
DIGITAL TO ANALOG CONVERSION

DIGITAL-TO-ANALOG CONVERTER (DAC)

Converts digital values to analog outputs of either voltage or current

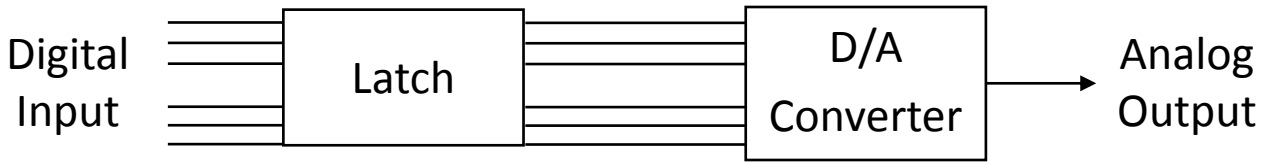
- Current DACs require external op-amp to convert back to voltage



DIGITAL-TO-ANALOG (D/A) CONVERSION

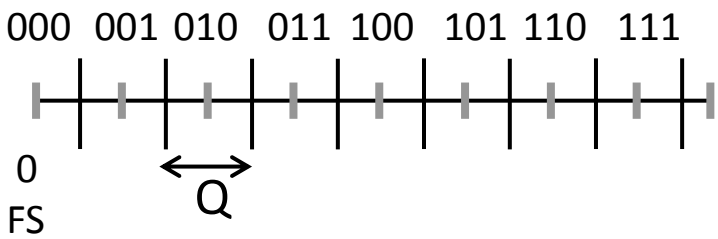
- Digital value is stored in a register (latch), then converted. Duration of conversion is called *settling time*.
- Output of the DAC remains the same until the next value is sent to the register (latch) – a zero-order hold.
- Basic concept:

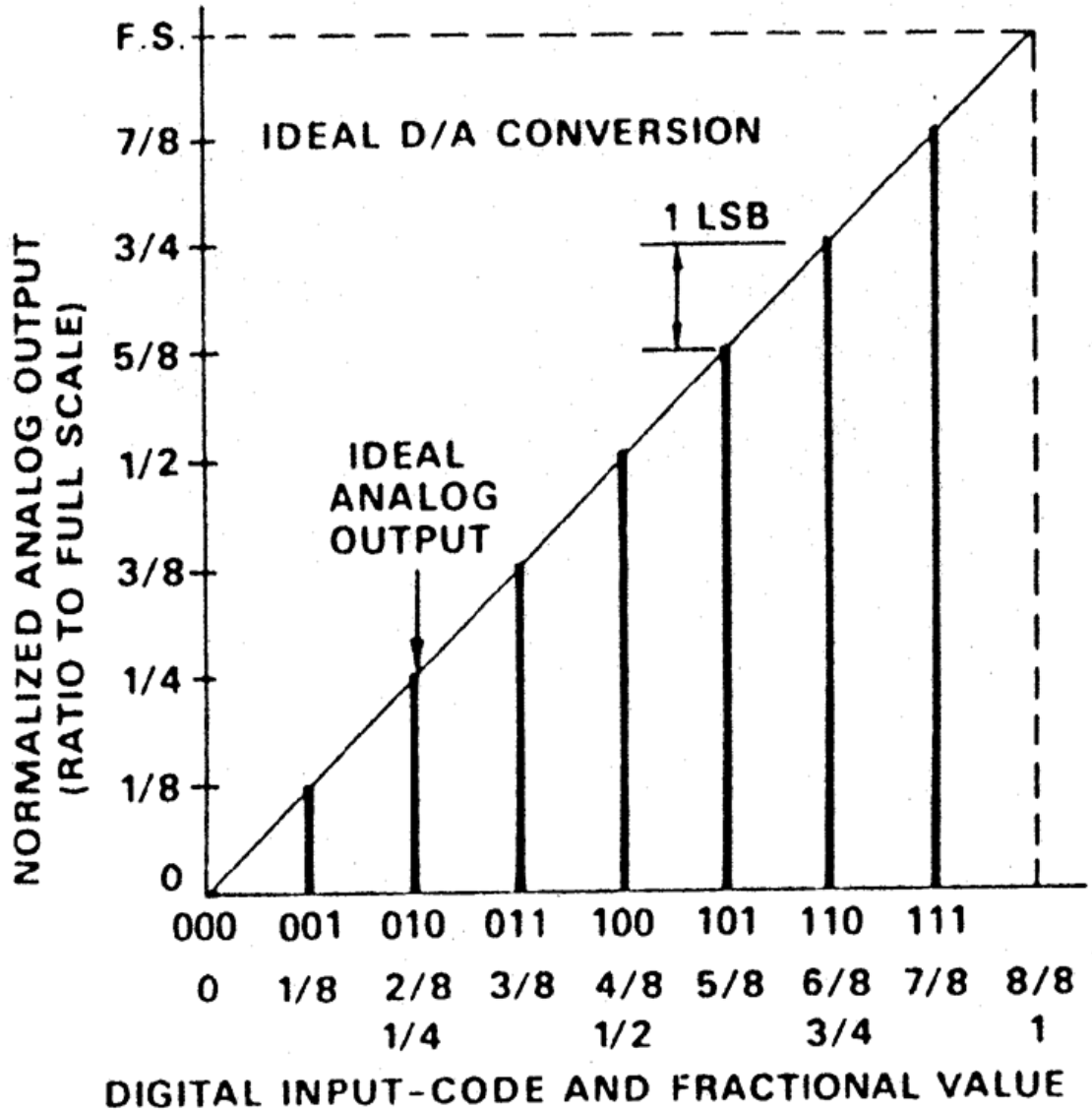
$$V_{OUT} = \underbrace{(b_{N-1} \cdot 2^{N-1} + b_{N-2} \cdot 2^{N-2} + \cdots + b_1 \cdot 2^1 + b_0 \cdot 2^0)}_{\text{Integer equivalent of binary code}} \cdot Q$$



DIGITAL-TO-ANALOG (D/A) CONVERSION

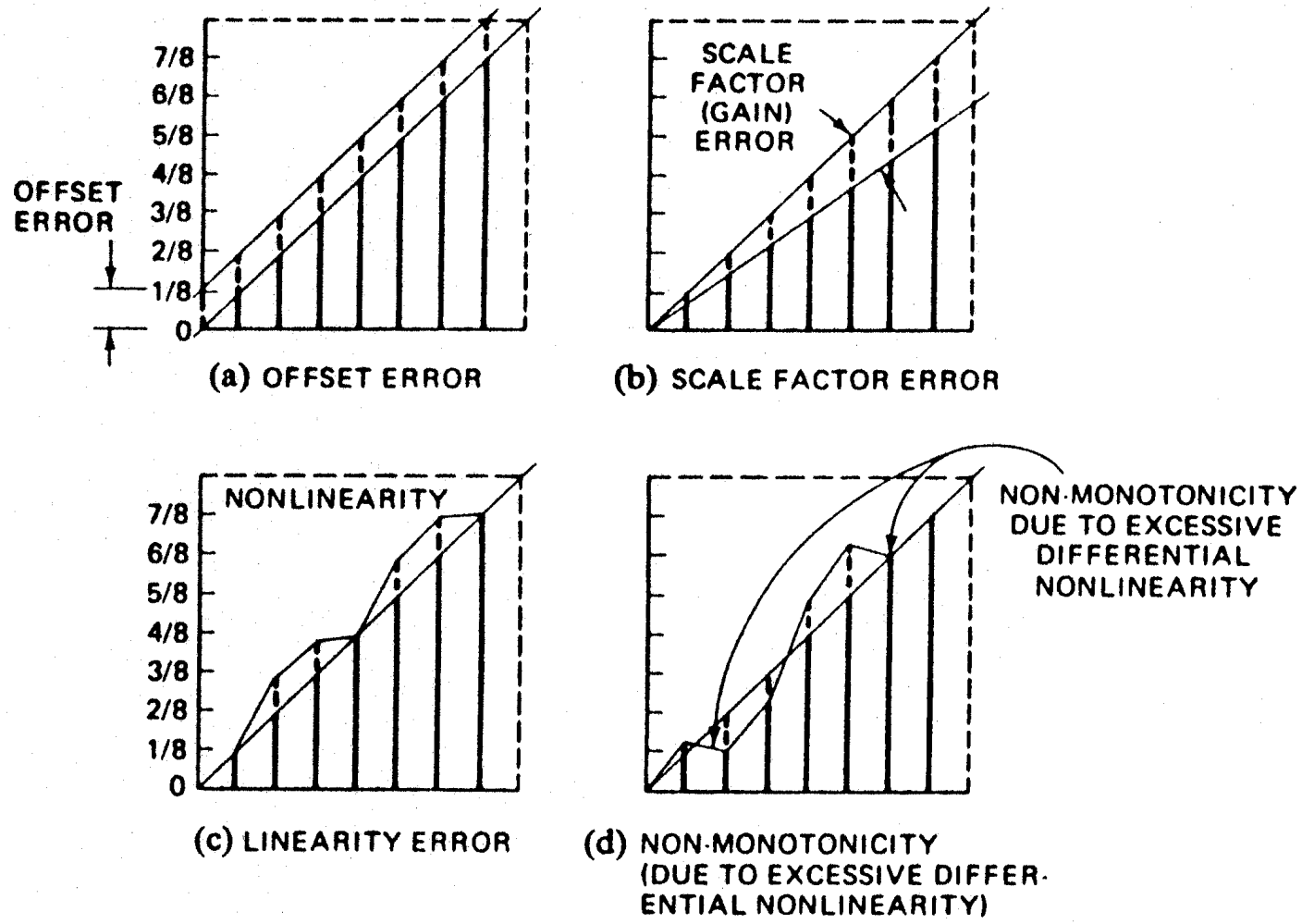
Ideal DA Conversion:





DIGITAL-TO-ANALOG (D/A) CONVERSION

DA Conversion Errors:



DIGITAL-TO-ANALOG (D/A) CONVERSION

DA Conversion Errors:

- Integral Non-linearity (INL): Max. deviation of analog output from ideal line
- Differential Nonlinearity (DNL): Max. deviation of analog output between two adjacent codes

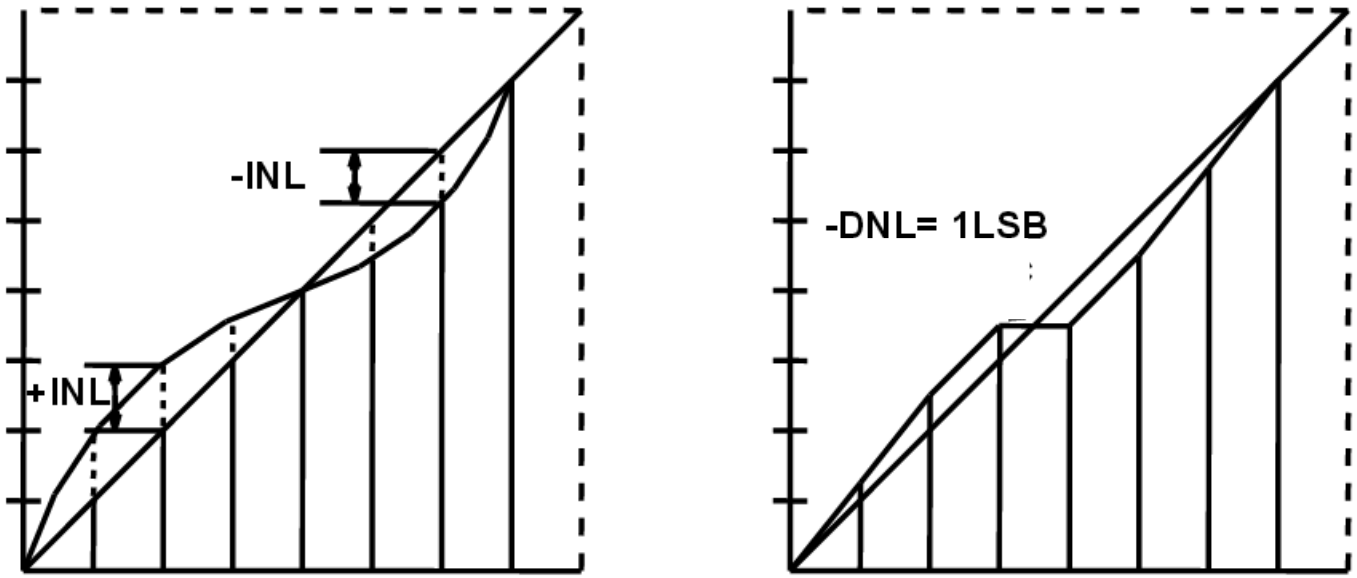


Image: Douglas A. Mercer, "Digital to Analog Converter Design," 2008

DIGITAL-TO-ANALOG (D/A) CONVERSION

Screenshot from Mouser.com website…

Digital to Analog Converters - DAC

D/A

☐ Stocked
☐ RoHS Compliant

Category ☑ Semiconductors > ☑ Data Converter ICs > ☑ Digital to Analog Converters - DAC

Applied Filters
☑ **Packaging** = Tube

Available Filters 2,936 Matches

Manufacturer	Number of Converters	Resolution	Interface Type	Settling Time	Maximum Operating Temperature	Number of DAC Outputs	Mounting Style	Package / Case
Analog Devices Inc.	1	8 bit	2-Wire, I2C	0.011 us	+ 70 C	1	SMD/SMT	CDIP
Cirrus Logic	2	10 bit	3-Wire	11 ns	+ 75 C	2	Through Hole	CDIP N
Exar	3	12 bit	3-Wire, Microwire, QSPI, SPI	0.012 us	+ 85 C	3		CDIP SB-28
Intersil	4	13 bit	3-Wire, Serial	12 ns	+ 105 C	4		CDIP-16
Maxim Integrated	8	14 bit	3-Wire, SPI	13 ns	+ 125 C	8		CDIP-18
Microchip	12	16 bit	DSP, Microwire, QSPI, SPI, Serial	14 ns	+ 150 C	12		CDIP-20
NJR	32	17 bit	I2C	0.015 us		32		CDIP-24
Reset	Reset	Reset	Reset	Reset	Reset	Reset	Reset	Reset

Apply Filters Apply Filters

DIGITAL-TO-ANALOG (D/A) CONVERSION

Screenshot from Mouser.com website…

Buy Selected | Compare Selected | Compare up to 20 parts. | Page: 1 2 3 4 5 6 7 8 9 10 … 118 Next

Select	Image	Mouser Part #	Mfr. Part #	Mfr.	Description		Availability	Pricing (USD)	Quantity	RoHS	Number o Converter
☐	 Enlarge	595-DAC8812ICPW	DAC8812ICPW	Texas Instruments	Digital to Analog Converters - DAC 16-Bit Dual Serial Input Multiplying	Data Sheet	1,391 In Stock Alternative Packaging	1: $22.01 10: $20.36 25: $18.71 50: $18.27 100: View	Buy Min.: 1 Mult.: 1	RoHS Details	2
☐	 Enlarge	584-DAC8420ESZ	DAC8420ESZ New At Mouser	Analog Devices	Digital to Analog Converters - DAC Quad 12B Serial Vout	Page 3,093 Data Sheet	508 In Stock Alternative Packaging	1: $47.08 5: $45.69 10: $44.31 25: $42.93 50: View	Buy Min.: 1 Mult.: 1	RoHS Details	4

DIGITAL-TO-ANALOG (D/A) CONVERSION

Basic types of D/A Converters:

- String
- Thermometer
- Weighted Resistor
- R/2R Ladder
- Multiplying
- Hybrid DACs (most IC devices)

DIGITAL-TO-ANALOG (D/A) CONVERSION

String DAC

- Fast, but requires large number of resistors and switches (one for each quantization level)
- Widely used in simple DACs

Only 1 switch closed at a time!

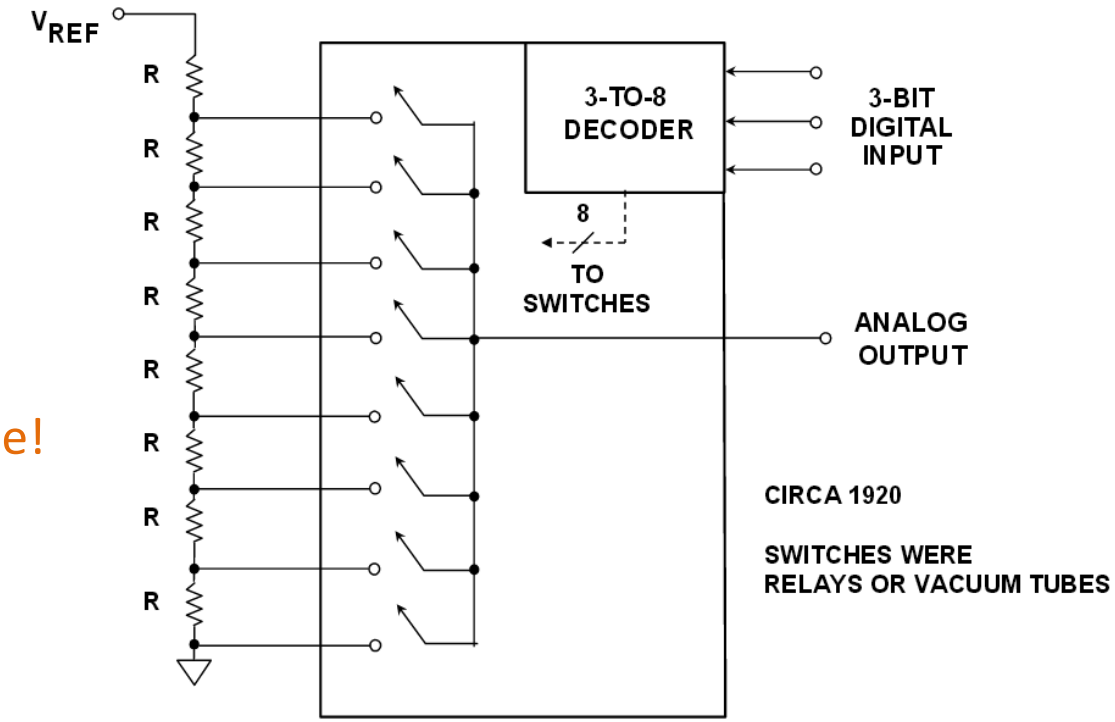
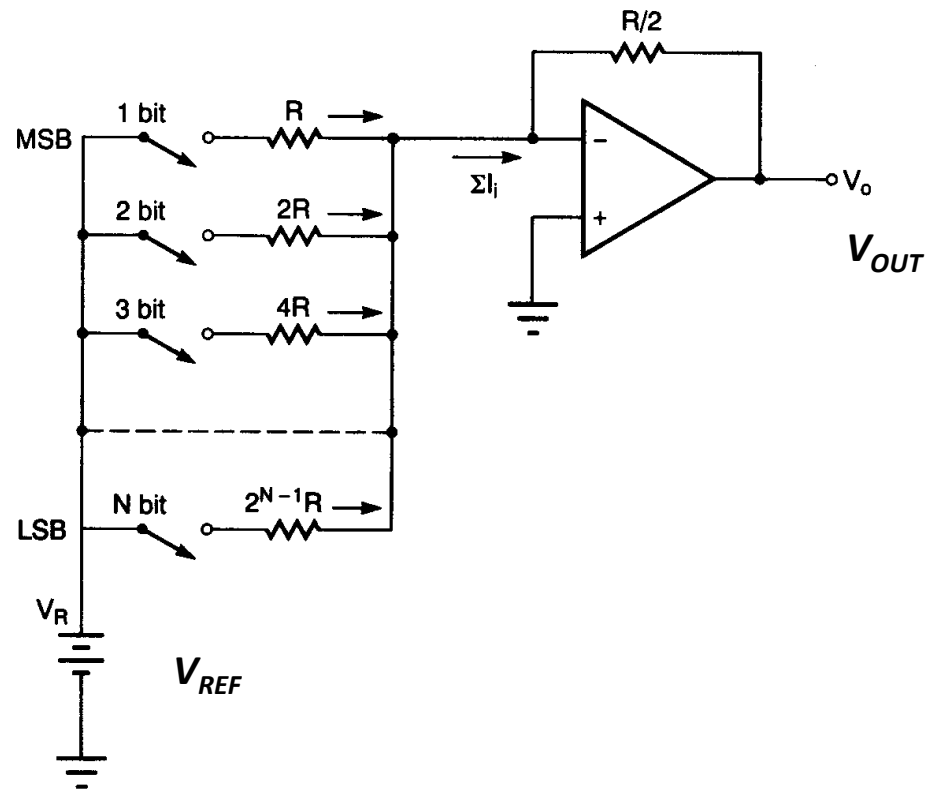


Image: Analog Devices MT-014 Tutorial

DIGITAL-TO-ANALOG (D/A) CONVERSION

Weighted (Scaled) Resistor DAC

- Fast, but not practical for high bit count due to expense of high precision resistors across a wide magnitude range
- Virtual ground at inverting op-amp input
- Requires:
  - Accurate reference voltage
  - Higher precision resistors



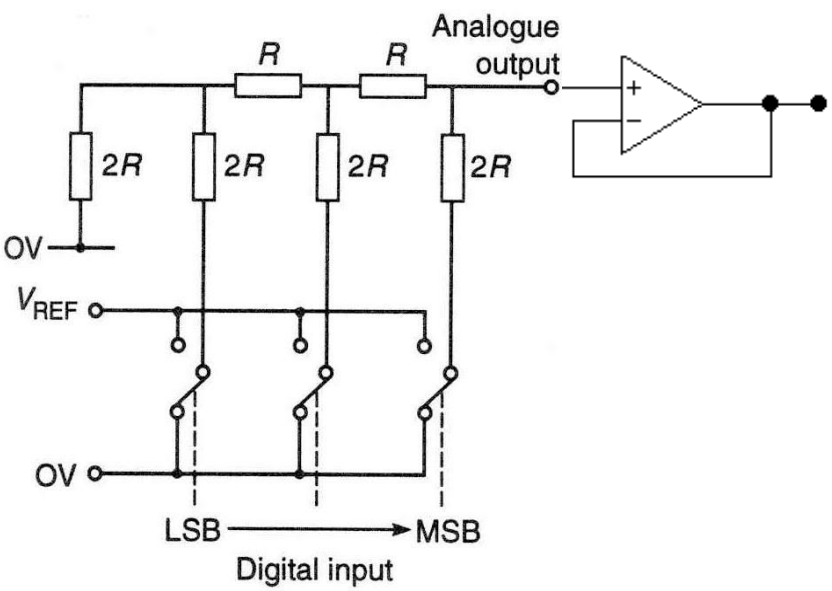
$$V_{OUT} = -\Sigma I_i \frac{R}{2} = -V_{\mathrm{REF}}\left(\frac{b_{N-1}}{2} + \frac{b_{N-2}}{4} + \cdots + \frac{b_0}{2^N}\right)$$

Can $|V_{\mathrm{OUT}}| = |V_{\mathrm{REF}}|$?

DIGITAL-TO-ANALOG (D/A) CONVERSION

R/2R Ladder DAC

- Uses just two resistance values (2R and R, closely matched)
- Input switches define a specific resistor divider network



If only MSB (100) is asserted:

$$V_{OUT} = \frac{4}{8} \cdot V_{REF} = \frac{1}{2} \cdot V_{REF}$$

If all bits are asserted (111):

$$V_{OUT} = \frac{7}{8} \cdot V_{REF}$$

$$V_{OUT} = (b_{N-1} \cdot 2^{N-1} + b_{N-2} \cdot 2^{N-2} + \cdots + b_1 \cdot 2^1 + b_0 \cdot 2^0) \cdot \frac{V_{REF}}{2^N}$$

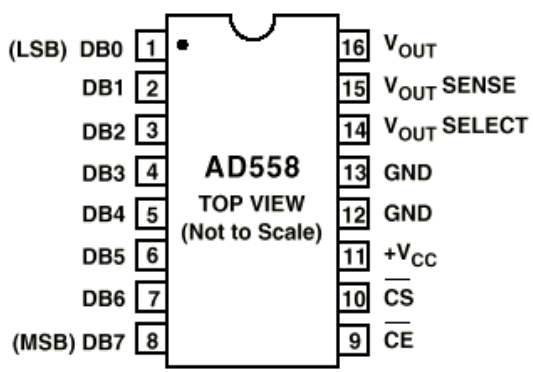
DIGITAL-TO-ANALOG (D/A) CONVERSION

Multiplying DAC

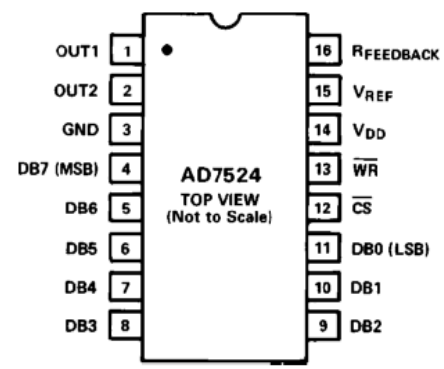
- Conventional DAC has internal reference voltage V_{REF} that is derived from the fixed power supply.
- Multiplying DAC has an externally supplied reference voltage.

$$V_{OUT} = (b_{N-1} \cdot 2^{N-1} + b_{N-2} \cdot 2^{N-2} + \cdots + b_1 \cdot 2^1 + b_0 \cdot 2^0) \cdot \frac{V_{REF}}{2^N}$$

- Advantages:
  - Use a constant frequency sinusoidal reference signal to achieve amplitude modulation, i.e. let $V_{REF} = V_R \sin(\omega t)$.
  - External V_{REF} can be precisely controlled to compensate for drift.



Internally Referenced



Multiplying DAC (Externally Referenced)

INTERFACING WITH A DAC

- Non-multiplying DAC – use AD558 (8 bit DAC) as example

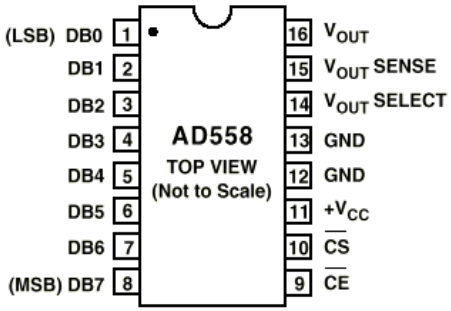
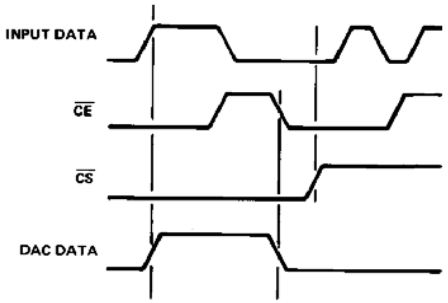
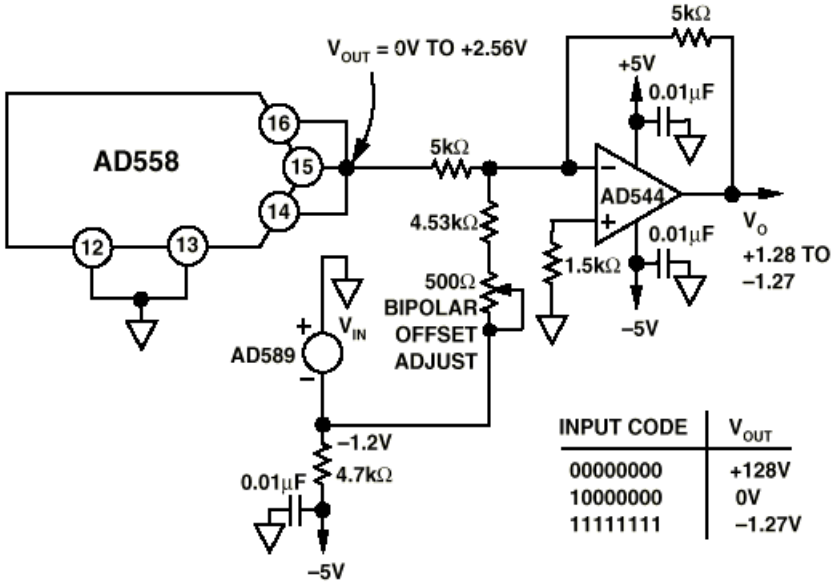


Table I. AD558 Control Logic Truth Table

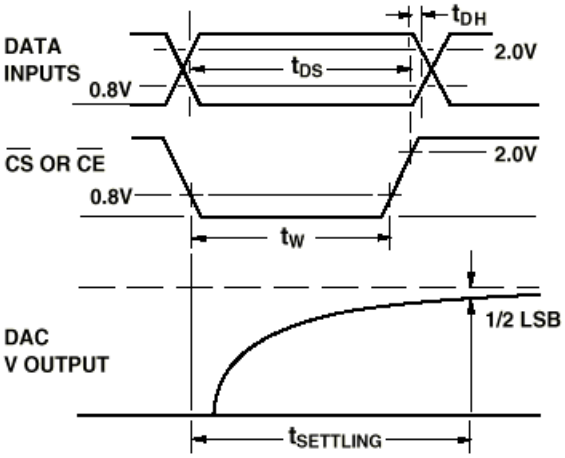
Input Data	$\overline{CE}$	$\overline{CS}$	DAC Data	Latch Condition
0	0	0	0	"Transparent"
1	0	0	1	"Transparent"
0	g	0	0	Latching
1	g	0	1	Latching
0	0	g	0	Latching
1	0	g	1	Latching
X	1	X	Previous Data	Latched
X	X	1	Previous Data	Latched

NOTES
X = Does not matter.
g = Logic Threshold at Positive-Going Transition.





INPUT CODE	V_{OUT}
00000000	+128V
10000000	0V
11111111	–1.27V



t_W = STORAGE PULSE WIDTH = 200ns MIN
t_{DH} = DATA HOLD TIME = 10ns MIN
t_{DS} = DATA SETUP TIME = 200ns MIN
$t_{SETTLING}$ = DAC OUTPUT SETTLING TIME TO ±1/2 LSB

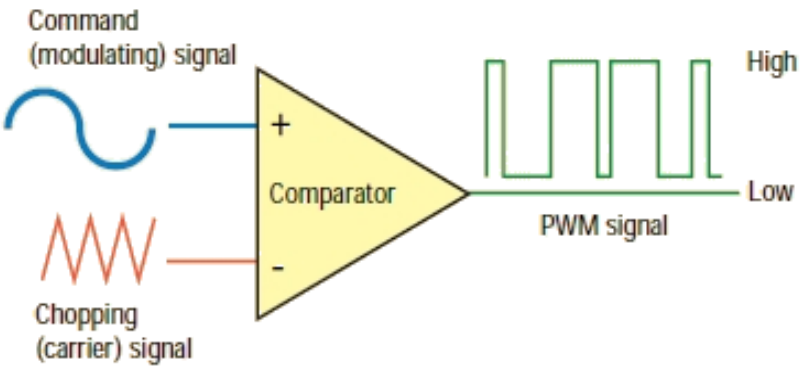
DIGITAL-TO-ANALOG (D/A) CONVERSION

Resistor methods rely on voltage dividers

- Many precision resistors necessary
- Wasted energy dissipated as heat

Pulse Width Modulation (PWM)

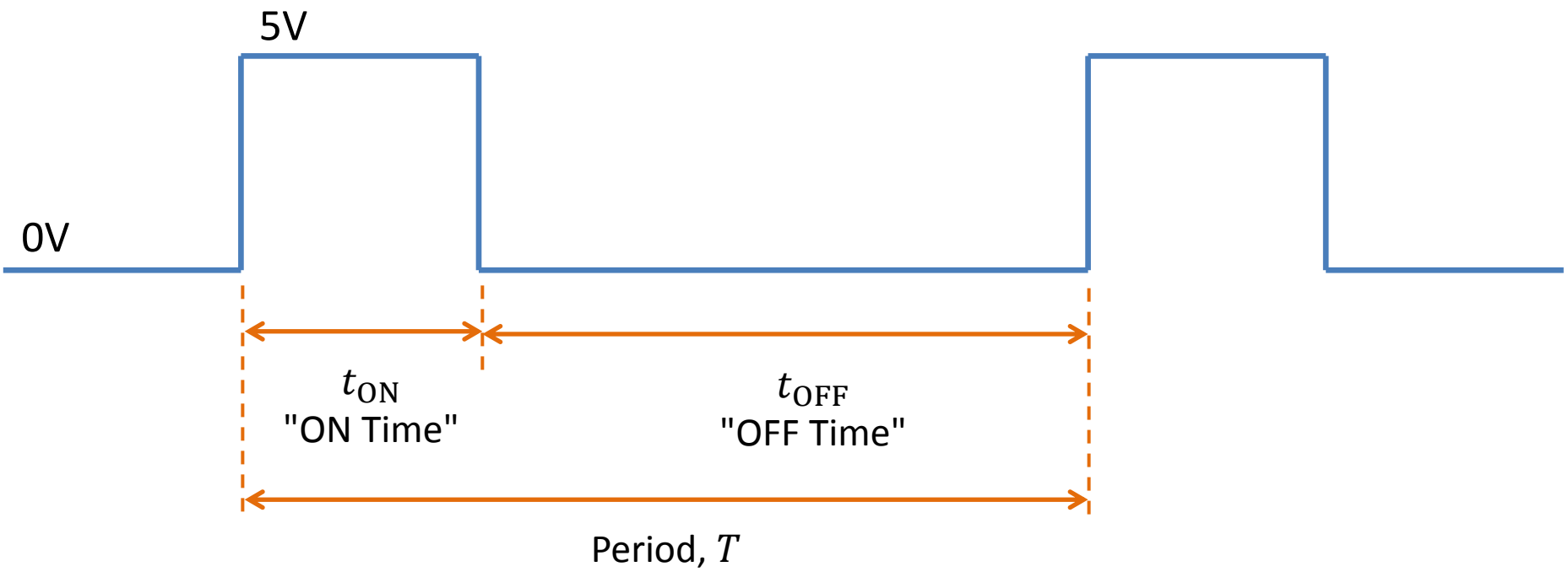
- Rectangular pulse wave
- Duty cycle controls average voltage
- Very high frequency content
- Need a low-pass filter to remove the sharp transitions at edges of the pulses!
- About 90% efficiency



PULSE WIDTH MODULATION (PWM)

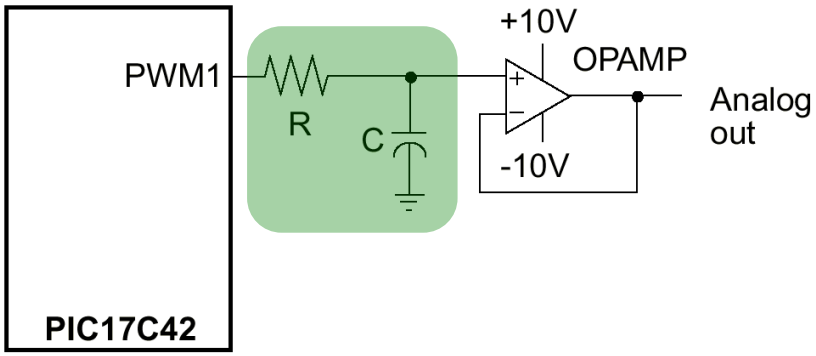
$$\text{Duty Cycle} = \frac{t_{ON}}{T} \Rightarrow 0 - 100\%$$

$$\text{Frequency (rad/sec)} = \frac{1}{T}$$



PULSE WIDTH MODULATION (PWM)

- Poor man's DAC
- Low pass filtering the PWM signal can produce an analog signal whose magnitude is proportional to the pulse width of the PWM signal
- For motor/motion control, the motor/motion system will act as the low pass filter
- Unipolar output
- Best suited when an analog output is needed but does not require a high resolution DAC



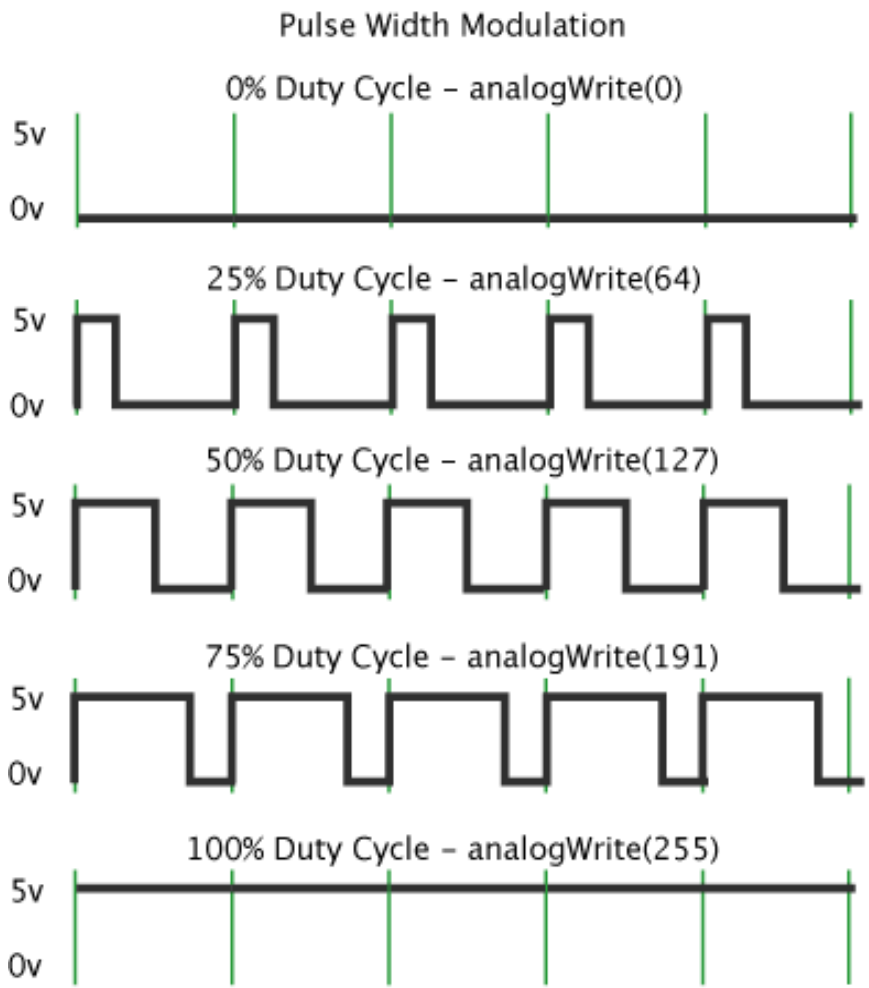
PULSE WIDTH MODULATION (PWM)

8-bit resolution:

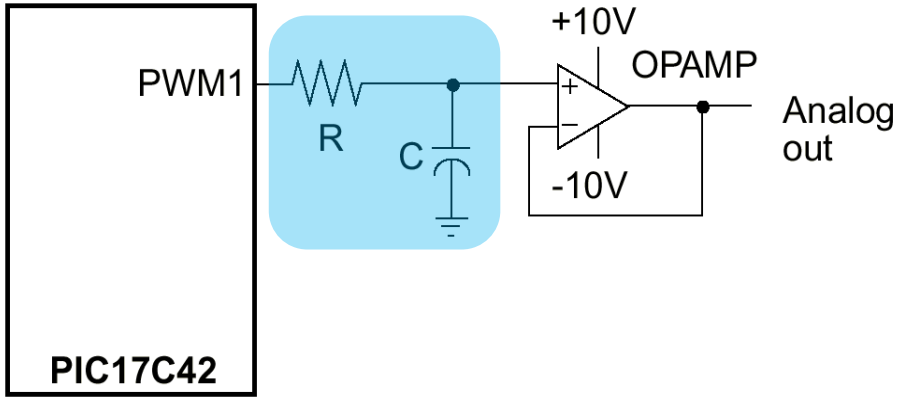
- 100%/255 => 0.39% per step
- 5V/255 => 19.6 mV per step

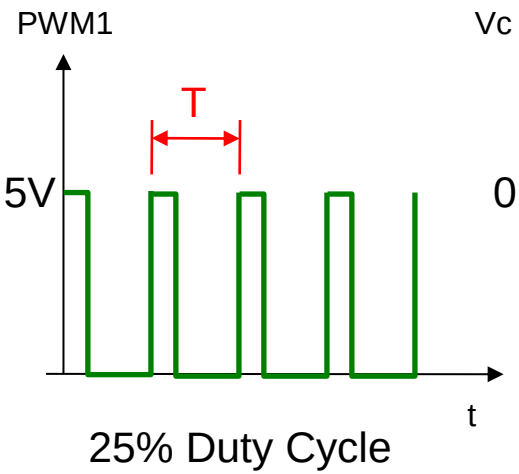
Arduino default PWM frequency:

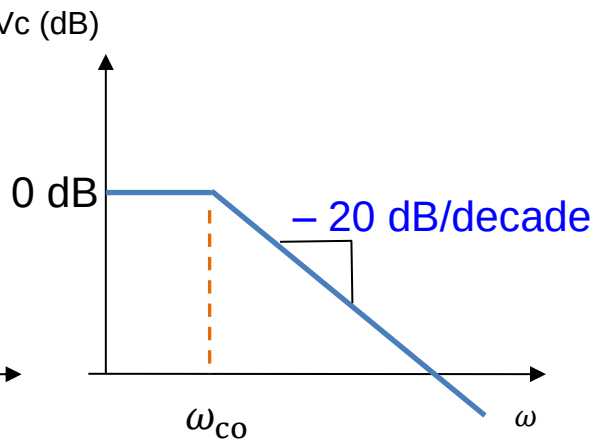
- Pins 5/6: ~976 Hz
- Pins 3/9/10/11: ~488 Hz
- Frequency can be increased to as much as 62.5 kHz by altering timer control registers
  - Pins 5/6: TCCR0B
  - Pins 9/10: TCCR1B
  - Pins 3/11: TCCR2B



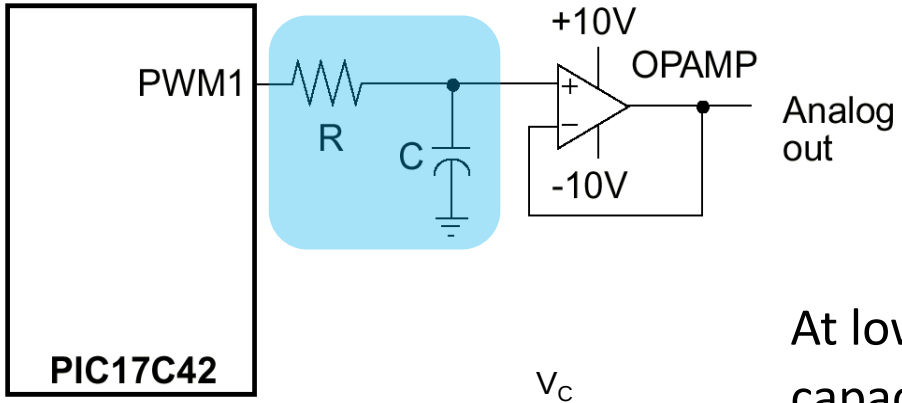
PULSE WIDTH MODULATION (PWM)

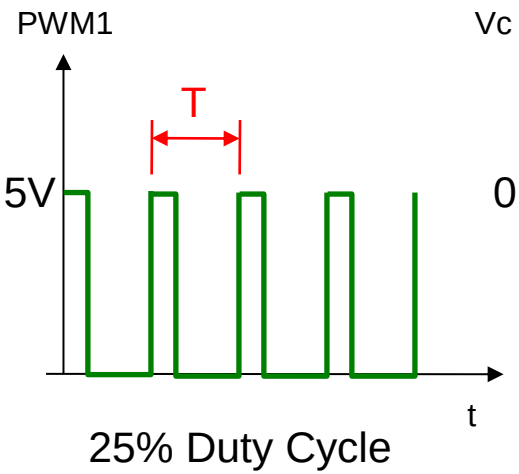


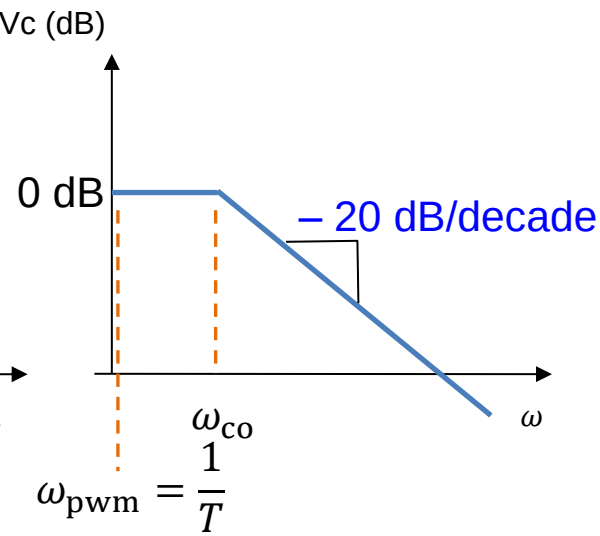




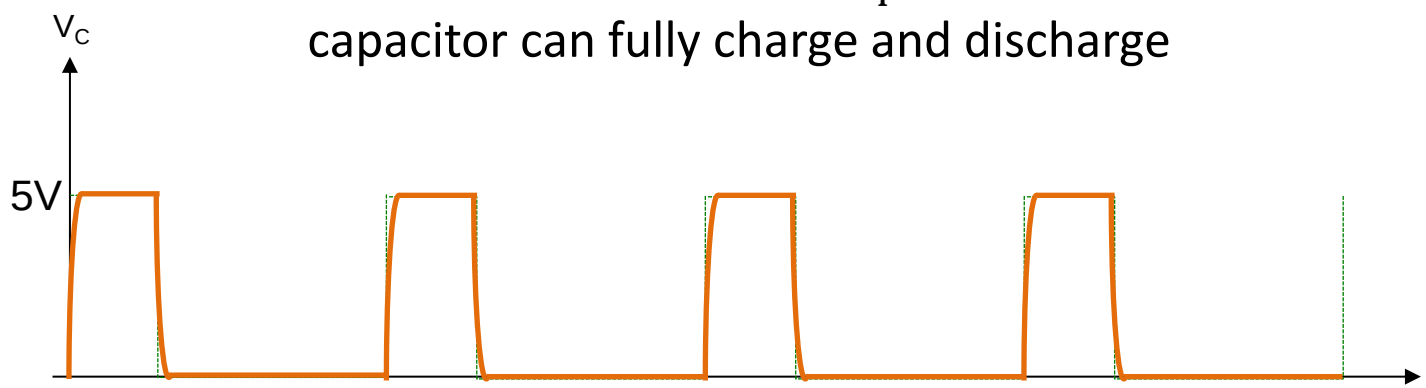
PULSE WIDTH MODULATION (PWM)



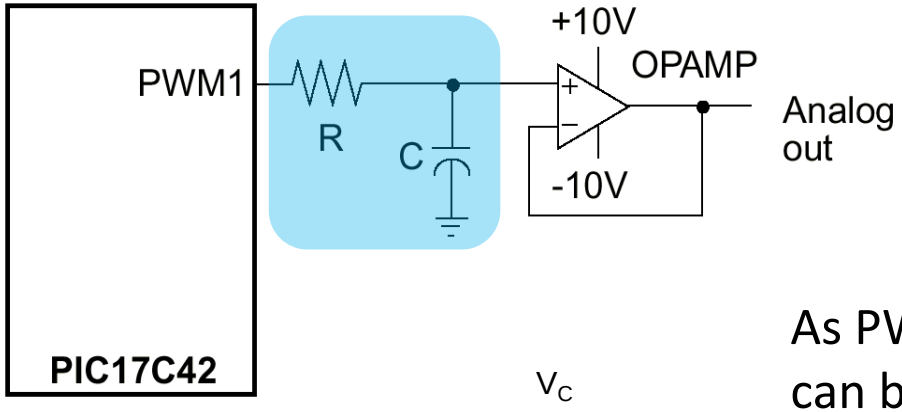


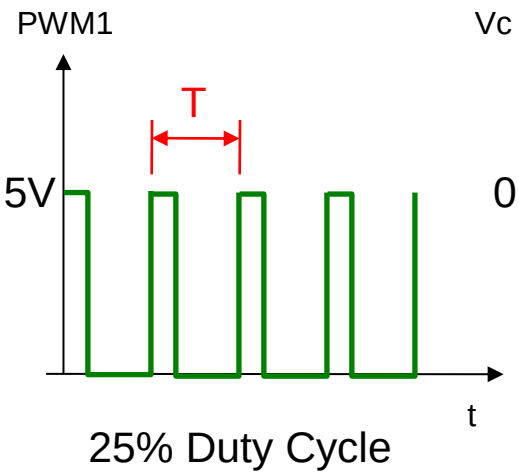


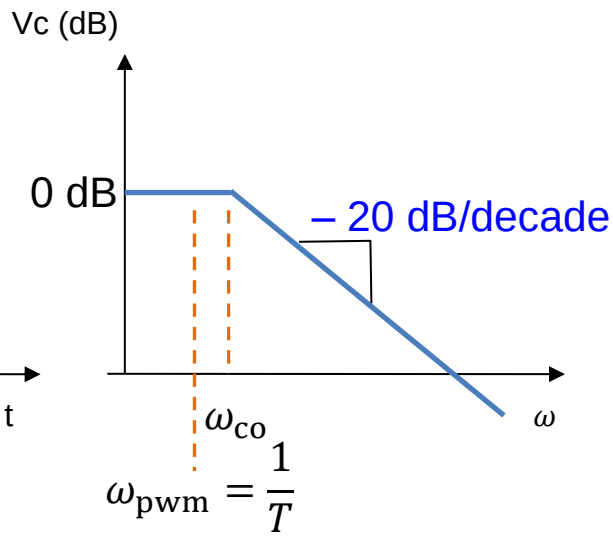
At low PWM frequency ($\omega_{pwm} \ll \omega_{co}$), capacitor can fully charge and discharge



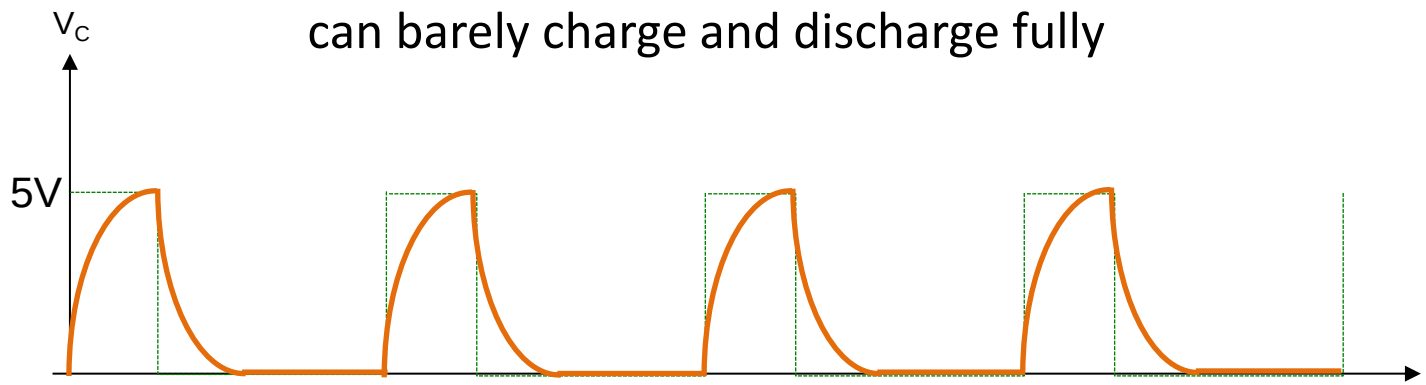
PULSE WIDTH MODULATION (PWM)



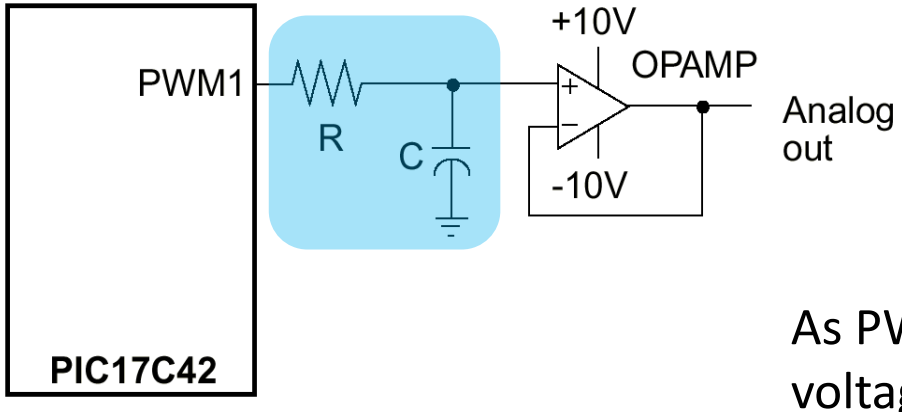


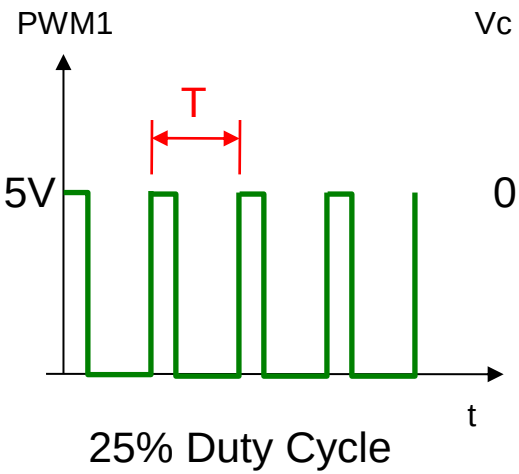


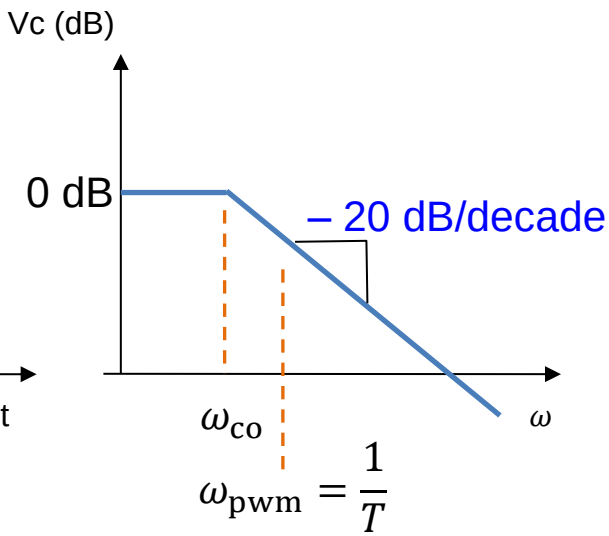
As PWM frequency increases, capacitor can barely charge and discharge fully



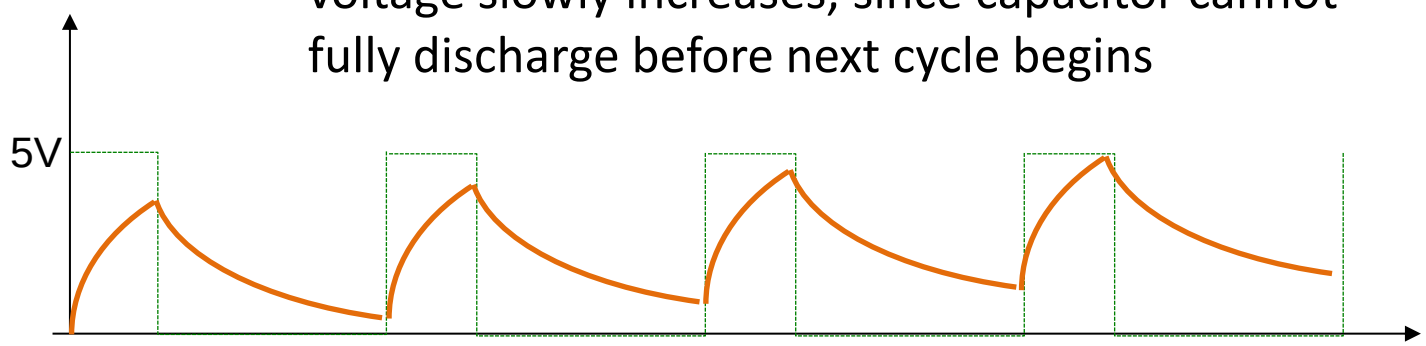
PULSE WIDTH MODULATION (PWM)



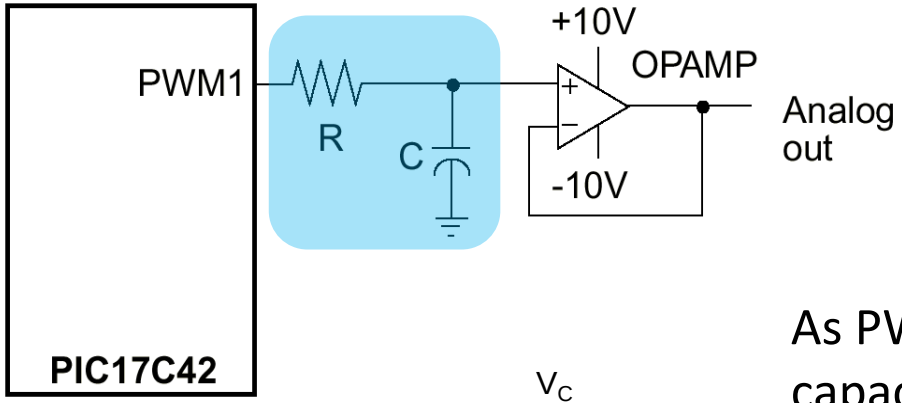


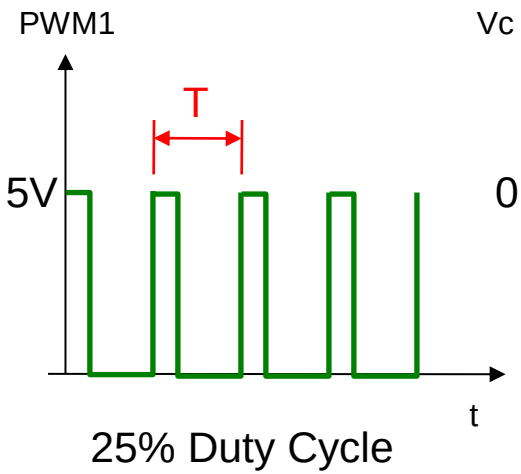


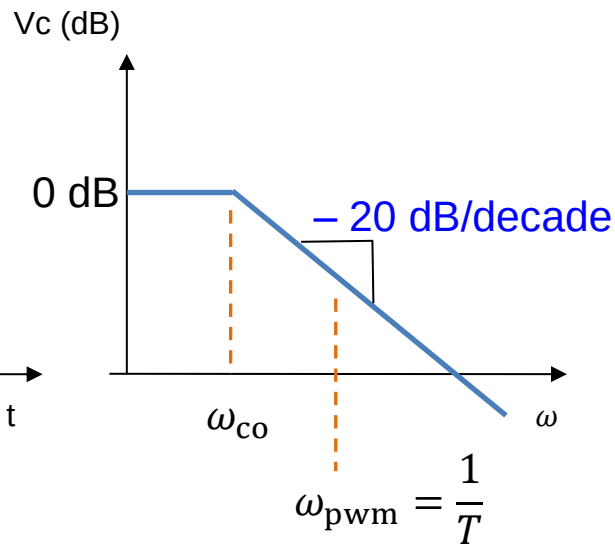
As PWM frequency increases further, capacitor voltage slowly increases, since capacitor cannot fully discharge before next cycle begins



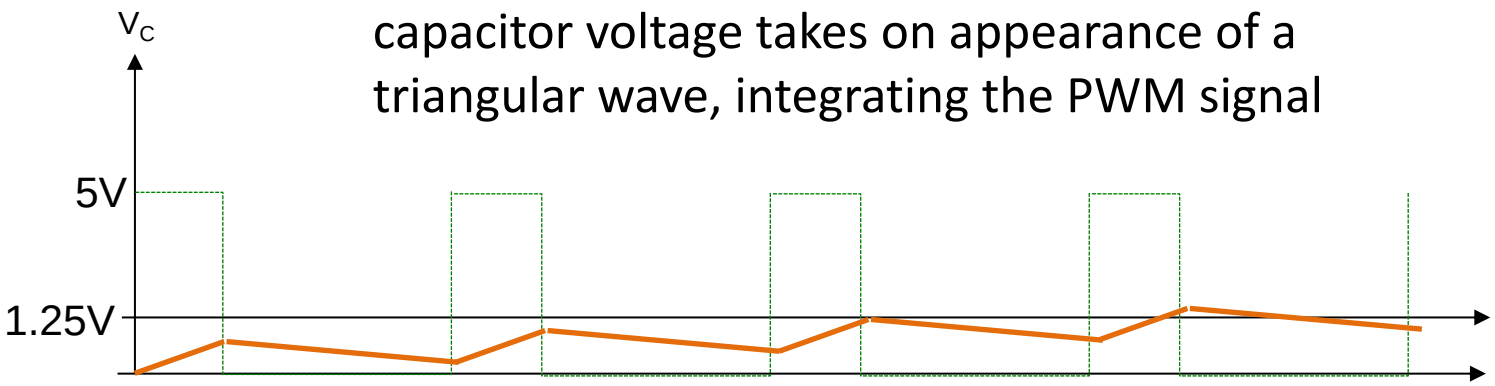
PULSE WIDTH MODULATION (PWM)







As PWM frequency increases even further, capacitor voltage takes on appearance of a triangular wave, integrating the PWM signal



PULSE WIDTH MODULATION (PWM)

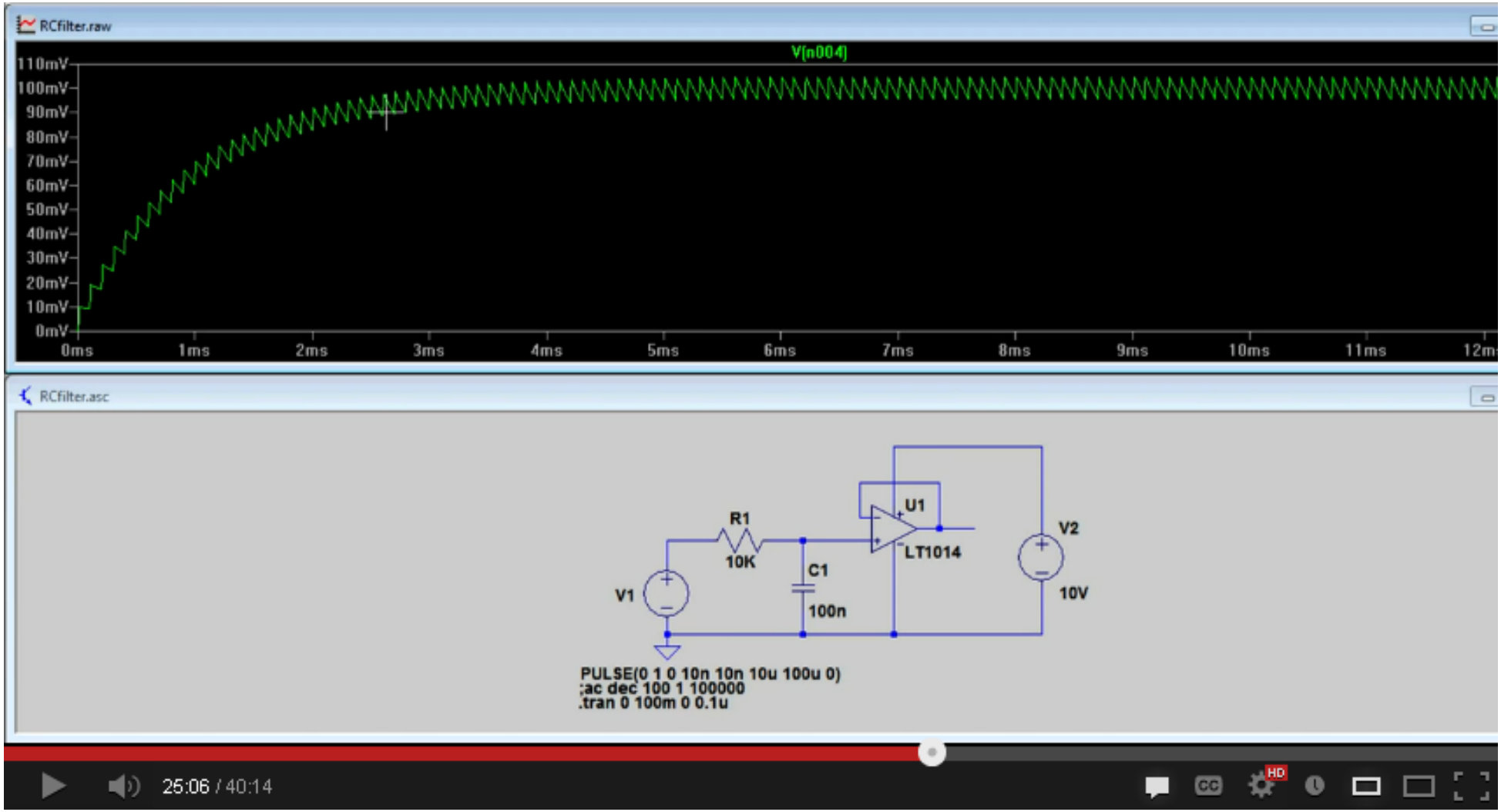
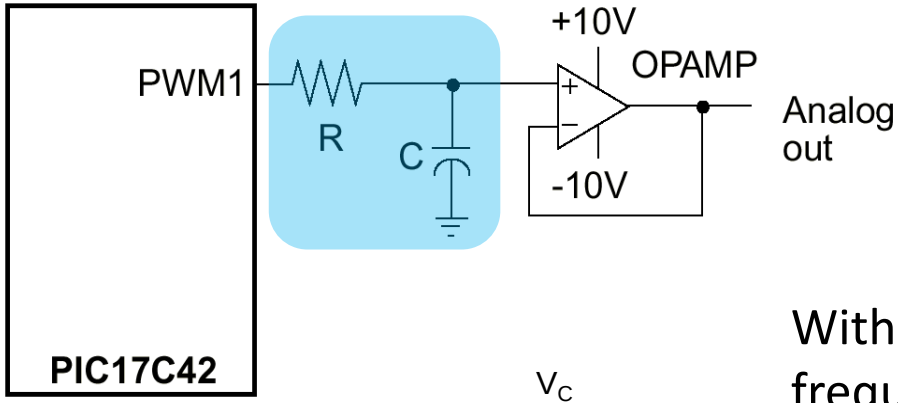
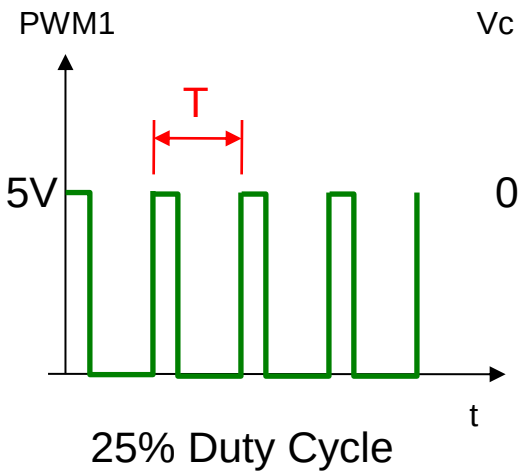
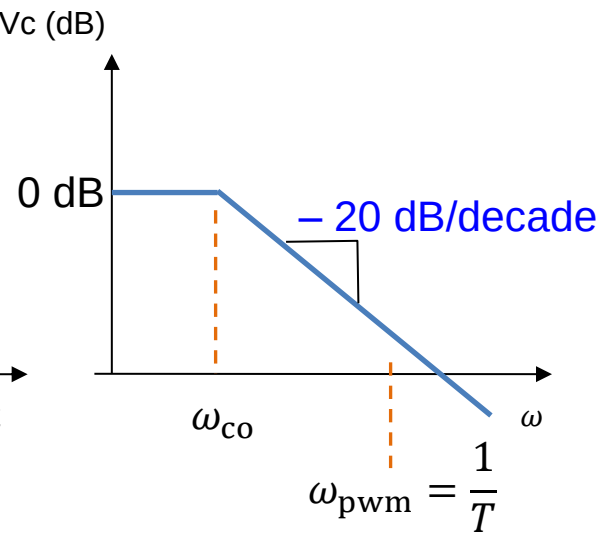


Image: http://www.youtube.com/watch?v=YaRDbw38x7Q

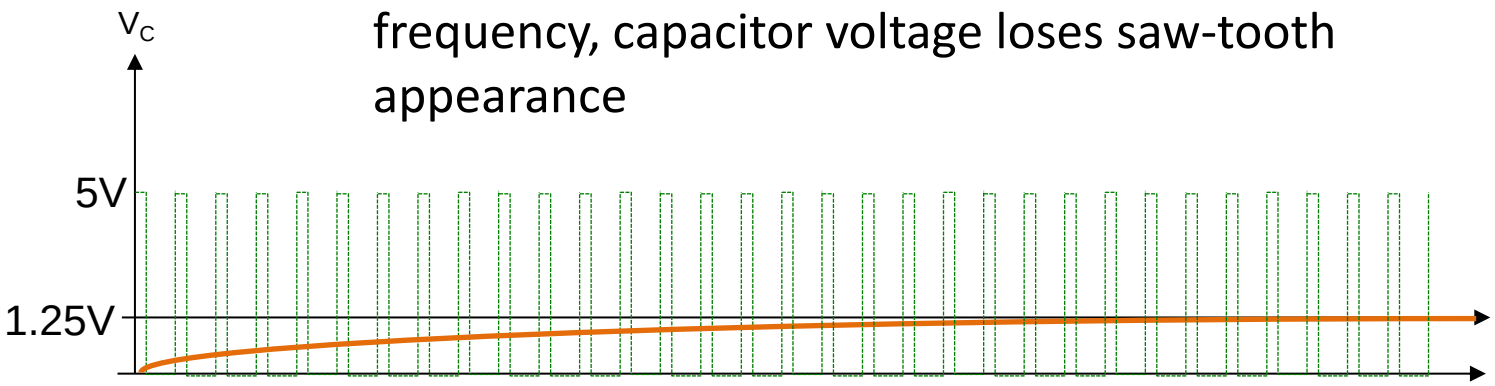
PULSE WIDTH MODULATION (PWM)





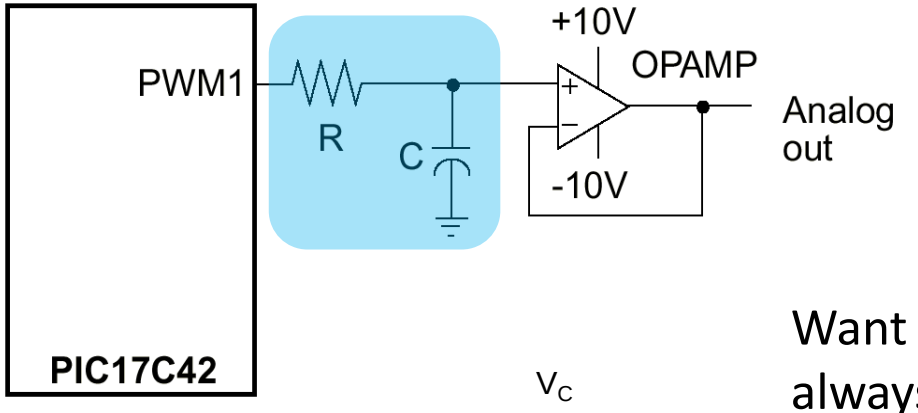


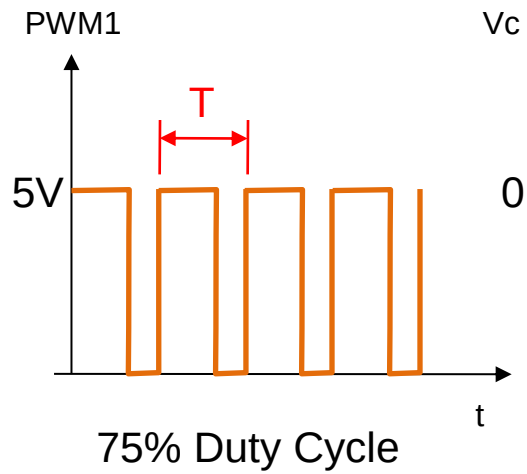
With PWM frequency well above cutoff frequency, capacitor voltage loses saw-tooth appearance

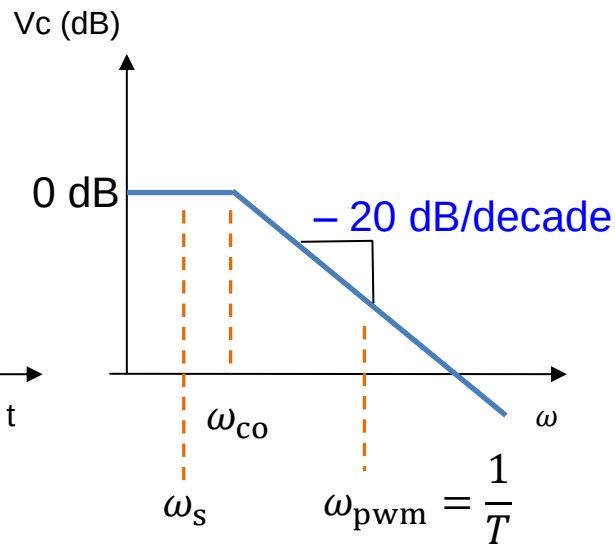


$V_{C,\text{ steady state}} = V_{\text{pwm}} \times \text{Duty Cycle \%}$

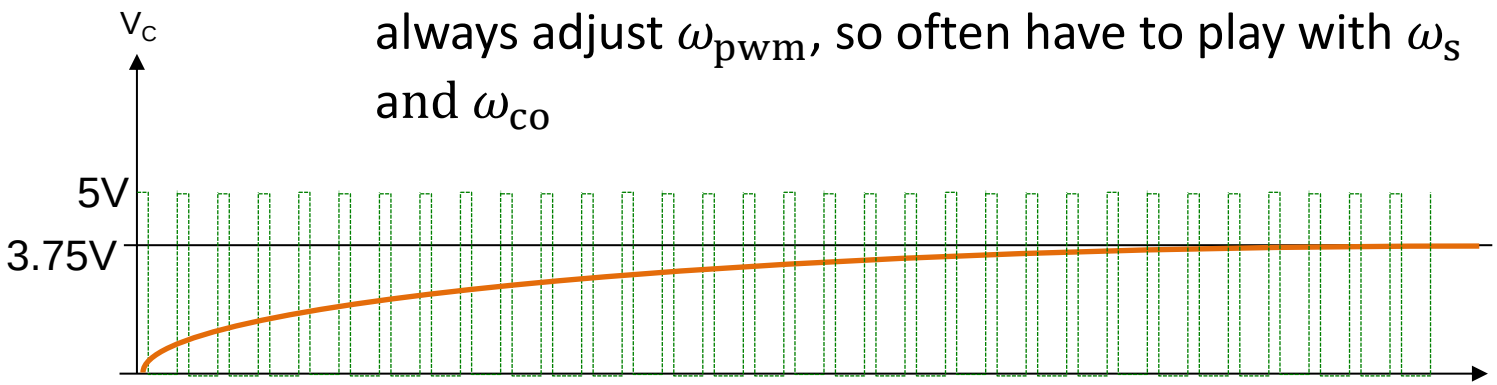
PULSE WIDTH MODULATION (PWM)







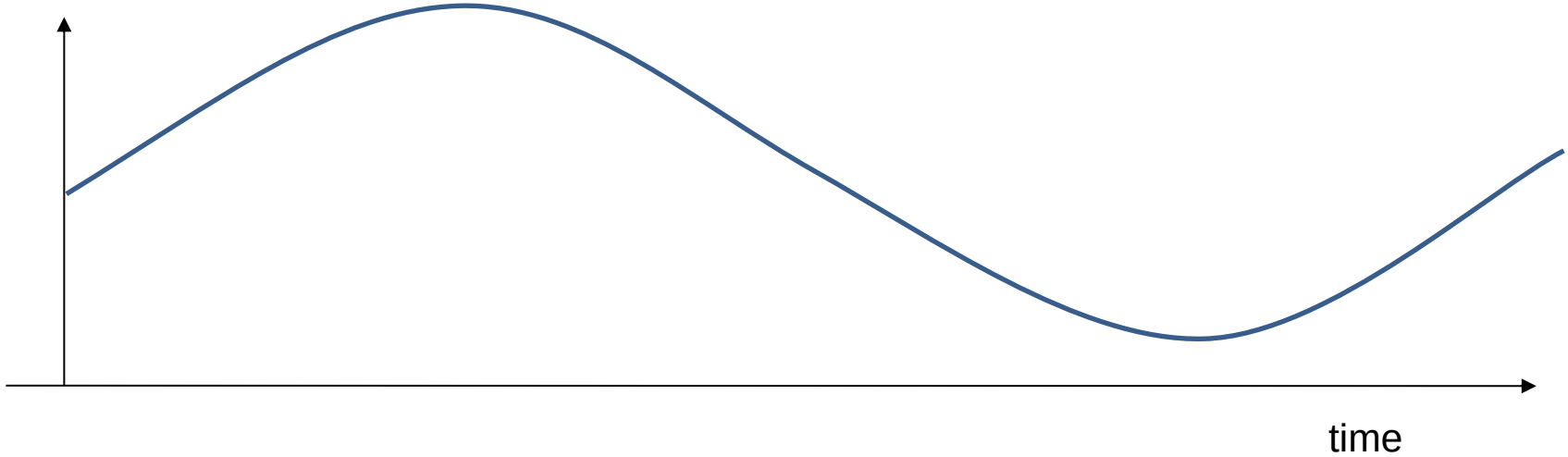
Want signal frequency ω_s less than ω_{co}. Can't always adjust ω_{pwm}, so often have to play with ω_s and ω_{co}



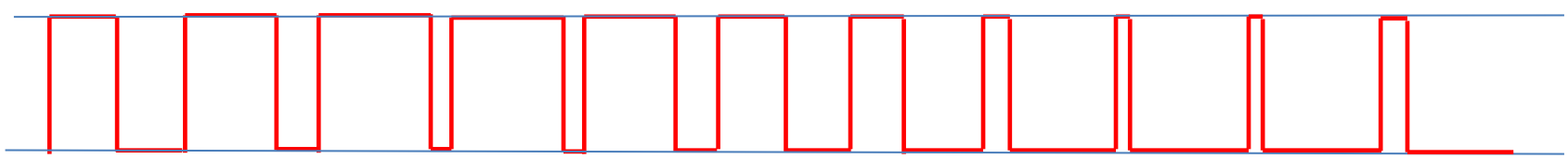
$V_{C,\text{ steady state}} = V_{pwm} \times$ Duty Cycle %

PULSE WIDTH MODULATION (PWM)

Digital Input



PWM Waveform



COMING UP…

Sensors and Actuators

- Types of sensors and actuators
- Interfacing with sensors and actuators
- DC Motors
- Stepper Motors