EEE3096S: Embedded Systems II

LECTURE 19:

ANALOGUE SIGNAL GENERATION AND DAC

Presented by:

STANLEY MBEWE

Electrical Engineering
University of Cape Town

OUTLINE OF LECTURE

- Recap: Timers, PWM use for signal generation
- Digital to Analogue Convertors (DACs)
  - GPIO'ed PWM DAC – Cheap and Nasty
  - PWM DAC – Better Method
  - R2R Ladder DAC
  - Binary Weighted DAC (BW-DAC)
- DAC Metrics

(brief recap slides on using timer hardware component to follow… you can use the DS3231 RTC for such things but it's cumbersome compared to an on-chip integrated timer)

Timers – brief recap

(sometimes also called ‘clocked triggers’ or ‘counter triggers’)

TIMER MODES OF OPERATION

- Normal Mode
  - Counts up, no clear, counter just wraps (signed or not)
- Clear Timer on Compare (CTC) match mode
- Fast Pulse Width Modulation (PWM) Mode
- Phase Correct PWM mode
- Phase and Frequency Correct PWM mode

PERIODIC INTERRUPT TIMER (PIT)

- Basically, same as a counter trigger:
  - 32-bit counter (commonly)
  - Set up by loading in start value (32-bit) *
  - Counter counts down with each clock pulse
  - Fixed clock source for PIT e.g. system clock
  - When timer value reaches zero
    - Generates interrupt
    - Reloads timer with start value

(alternate implementation, counter trigger: is that it may have one or more trigger value for which an interrupt is generated, the last trigger may make the counter reset)

* Calculating load value X_{count} : round($T*X_{count}$) = period, $T = 1/f_{CLK}$ s where f_{CLK} = clock frequency

Pulse Width Modulation (PWM)

(recap)

PWM: POWER MODERATING

- Most simple instance of PWM:
  - a means of changing the duty cycle of a pulse signal so as to change the net power, e.g. to dim a LED.

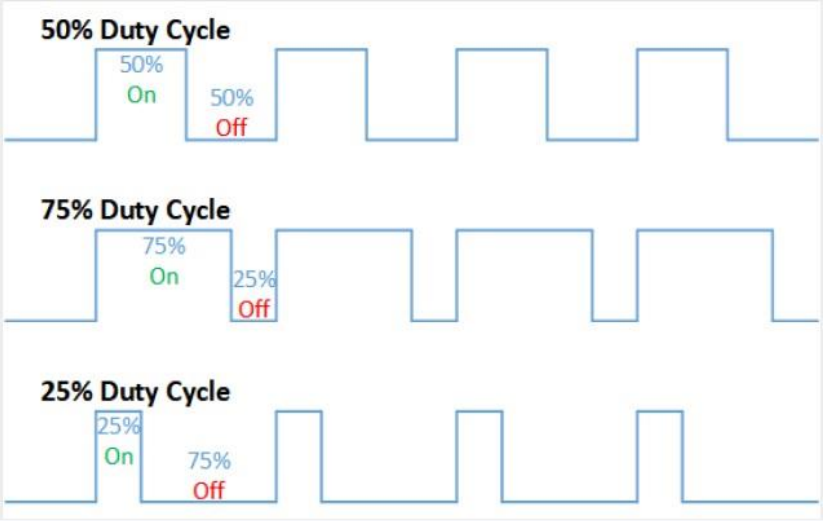


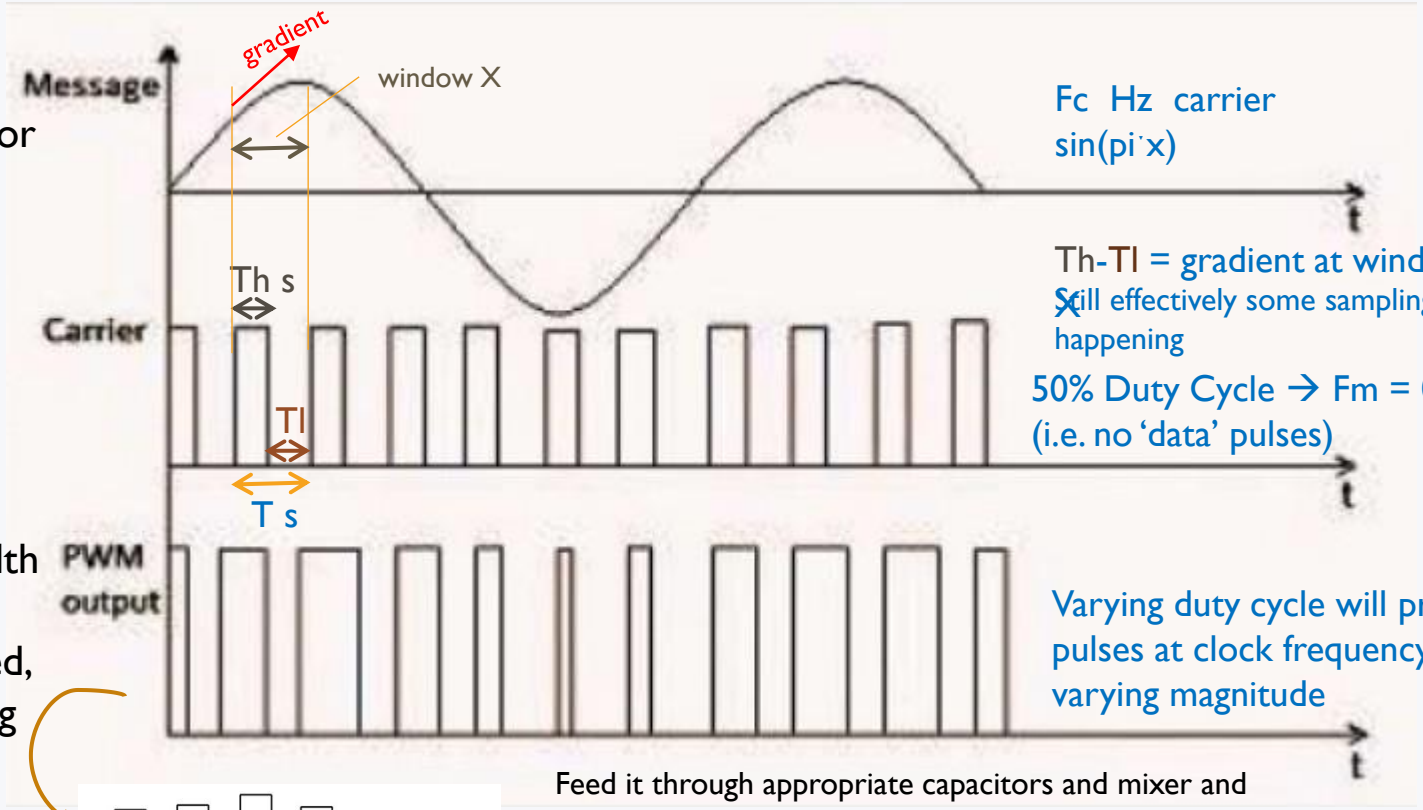
Image Source: https://upload.wikimedia.org/wikipedia/commons/b/b8/Duty_Cycle_Examples.png

PWM: COMMS USE IN MODULATION

Original message or signal

Carrier (clock)

Pulse width of clock modulated, containing data



Fc Hz carrier
sin(pi·x)

Th-Tl = gradient at window
Still effectively some sampling happening

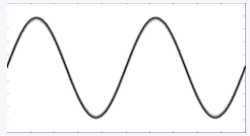
50% Duty Cycle → Fm = 0
(i.e. no 'data' pulses)

Varying duty cycle will produce pulses at clock frequency but varying magnitude

Vary Duty Cycle → Fm = pulses of varying V

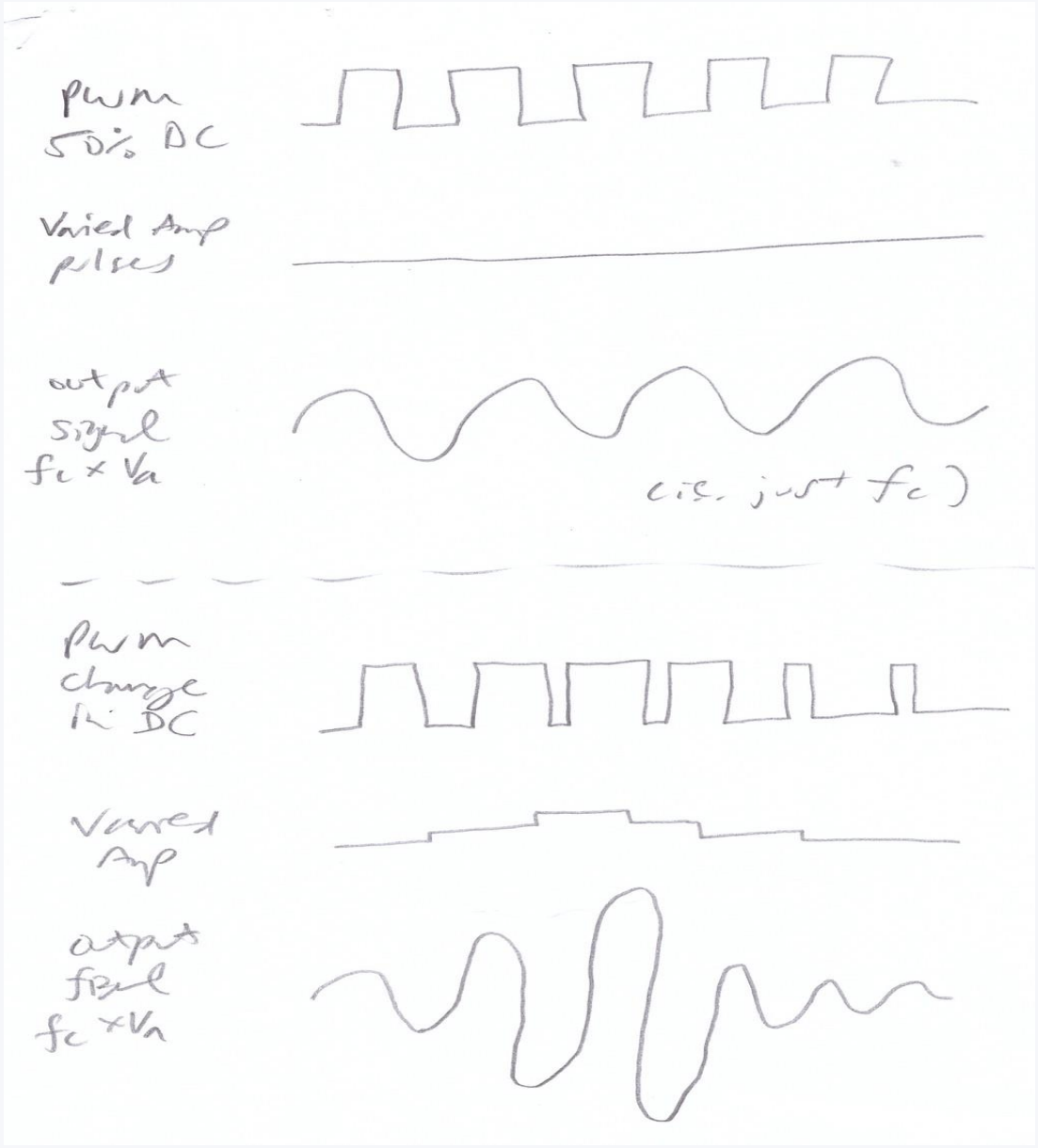
Feed it through appropriate capacitors and mixer and you can create a modulated analogue signal to send

Fm Hz = 1/(K·T)



* In some Software Defined Radio (SDR) designs

PWM TO MODULATED CARRIER



This is an illustration of the approach of using a PWM signal, generated from a microprocessor, that is converted into a varied amplitude pulse train, to modulate a carrier signal.

At the top we see a PWM of duty cycle 50% for the duration, just causes the same carrier to be output.

If we vary the PWM, this translates into a pulses of varied amplitude, which then amplitude modulates the produced carrier signal.

Digital to Analogue Convertors (DACs)

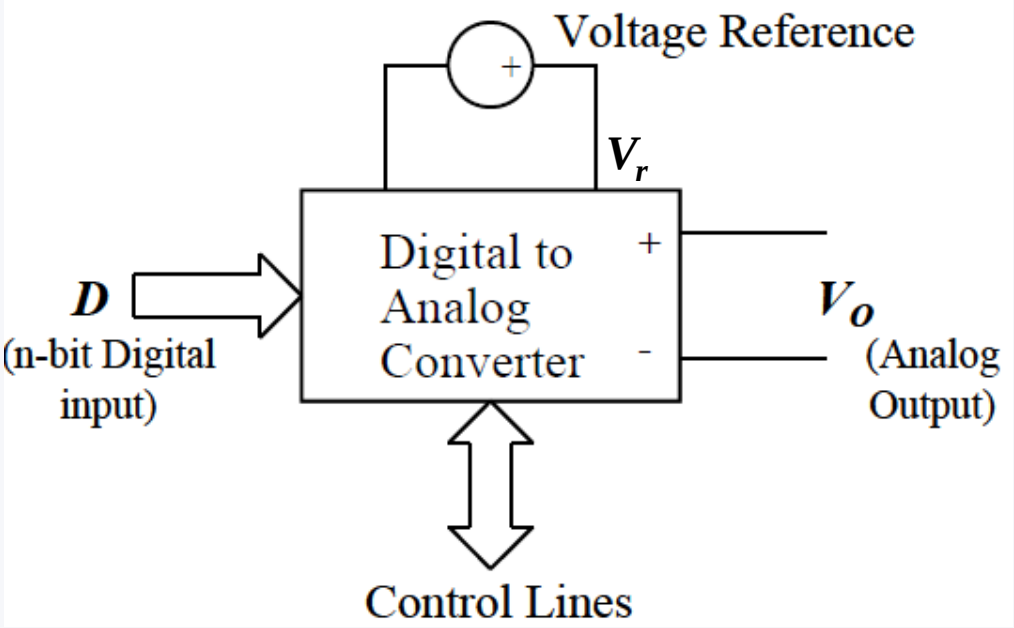
Now that we know the PWM and the PIT we can get into the DAC …

these acronyms alas don't rhyme but these techniques work so well together that one wishes the terms did rhyme!

DAC CONCEPTUAL DESIGN

About converting a digital binary value into a voltage…

- Three characteristics of a DAC
  - D is the digital input
  - V_O is the Analog output
  - V_r is a precise, stable, known voltage reference
- Some control lines: e.g. activate DAC to convert the input signal.

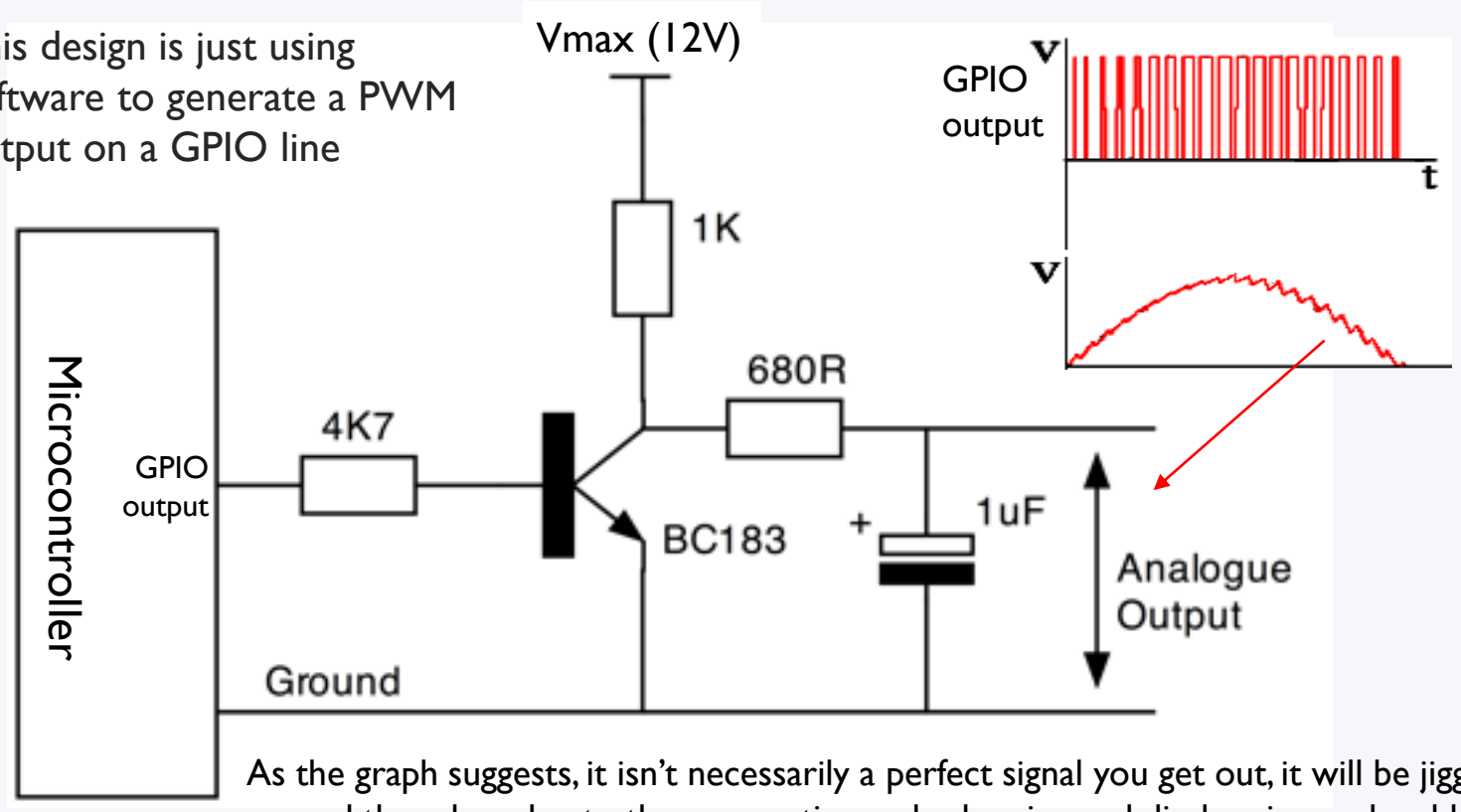


- Objective of a DAC is to turn a digital value into a proportional Voltage or Current (according to a transfer function, i.e. mapping what V or I must be produced for a given value)

GPIO'ED PWM DAC – CHEAP AND NASTY

This hardly needs explaining: basically you change the duty cycle of a PWM digital signal (best practice) that is then sent through a buffer or amp and a simple LPF capacitor circuit:

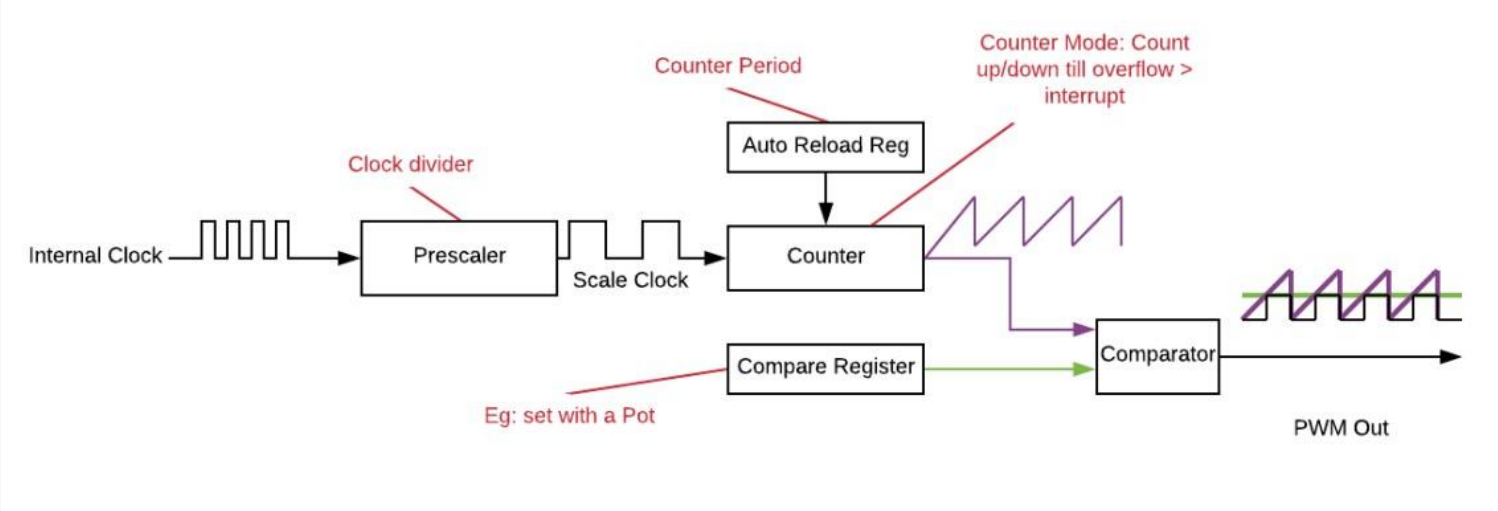
This design is just using software to generate a PWM output on a GPIO line



As the graph suggests, it isn't necessarily a perfect signal you get out, it will be jiggered around the edges due to the cap continuously charging and discharging, and sudden big voltage changes (e.g. from 0000_2 → 1111_2 form one sample to the next may be impossible.

PWM DAC – BETTER METHOD

This is a more elegant approach (a bit more elegant, not perfect!), and likely what you'd find in a cheap but slow DAC chip. It makes use of a counter trigger timer to generate a PWM signal that can then be fed into the capacitor circuit (see previous slide) to create a voltage level output.



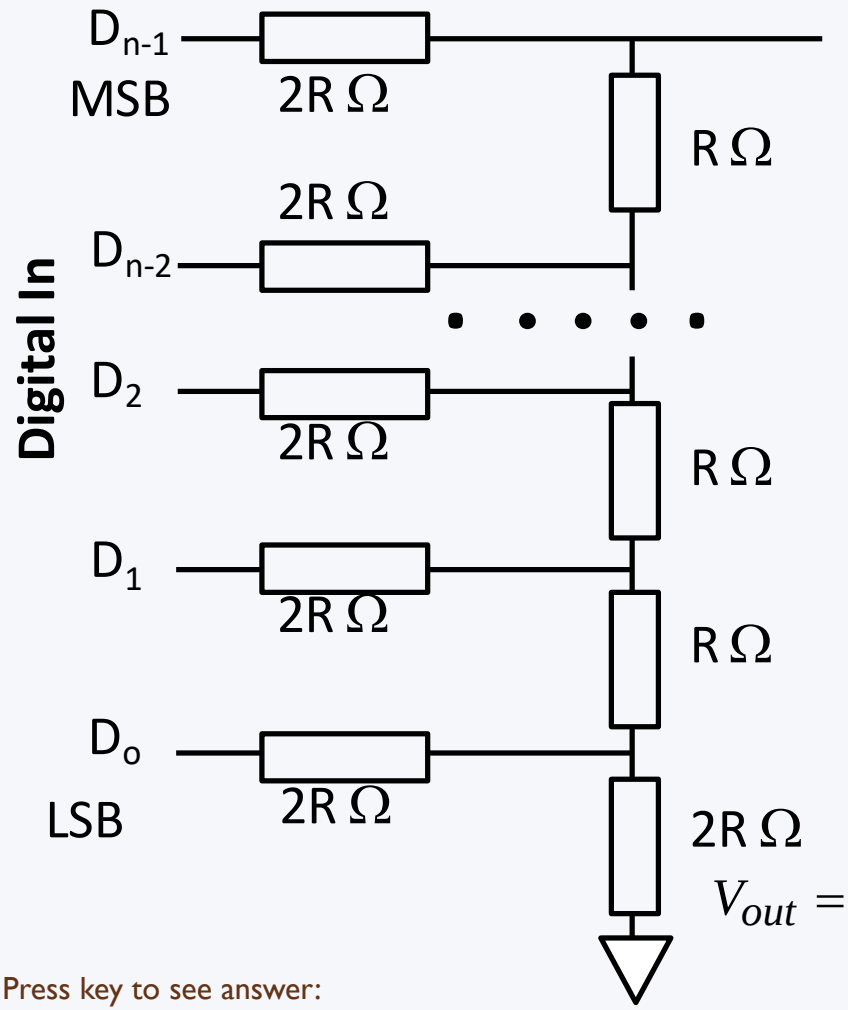
There's a calculation of duty cycle → voltage, which may be very temperamental (e.g. due to temperature, humidity etc. changes in the environment).
A start-up calibration process may be needed by which a voltage regulator would provide one (or more) high quality voltage references and comparator by which to calibrate the DAC. This may need to be done every couple 100ms because as the device runs, tasks change, and time moves on, conditions may change.

[5]

PWM DAC OPERATION

- Operation:
  - Pass PWM signal into a buffer or AMP and a LPF to get an equivalent Voltage
  - Changing the duty cycle produces the required voltage
- Resolution:
  - How many bits in your PWM generating counter gives you number of distinct analogue voltages to generate.
- Advantages:
  - Cheap and easy to implement
- Disadvantages:
  - Very noisy, need much filtering for many applications
  - Speed dependant on size of cap (also impacts noise)
  - Can be highly susceptible to environmental conditions

R2R LADDER DAC



V_{out} – Analog Out

Resistance ladder: This is how you can build one from first principles. It is pretty much efficient and a typical approach to implement a DAC. It is well-suited to being fabricated as a silicon chip. Assume the D values are either 0 or 5V, there are things like FETs hidden.

Check what you get for Vout given n=4, D_i inputs set to 1V or 0V – e.g. see what 1001_2 will output, it's a good test of your knowledge of Thévenin's theorem:

$$V_{out} = \frac{V_r}{2^n}\left(D_{n-1}2^{n-1} + D_{n-2}2^{n-2} + \ldots + D_o 2^0\right)$$

Press key to see answer:

Answer:

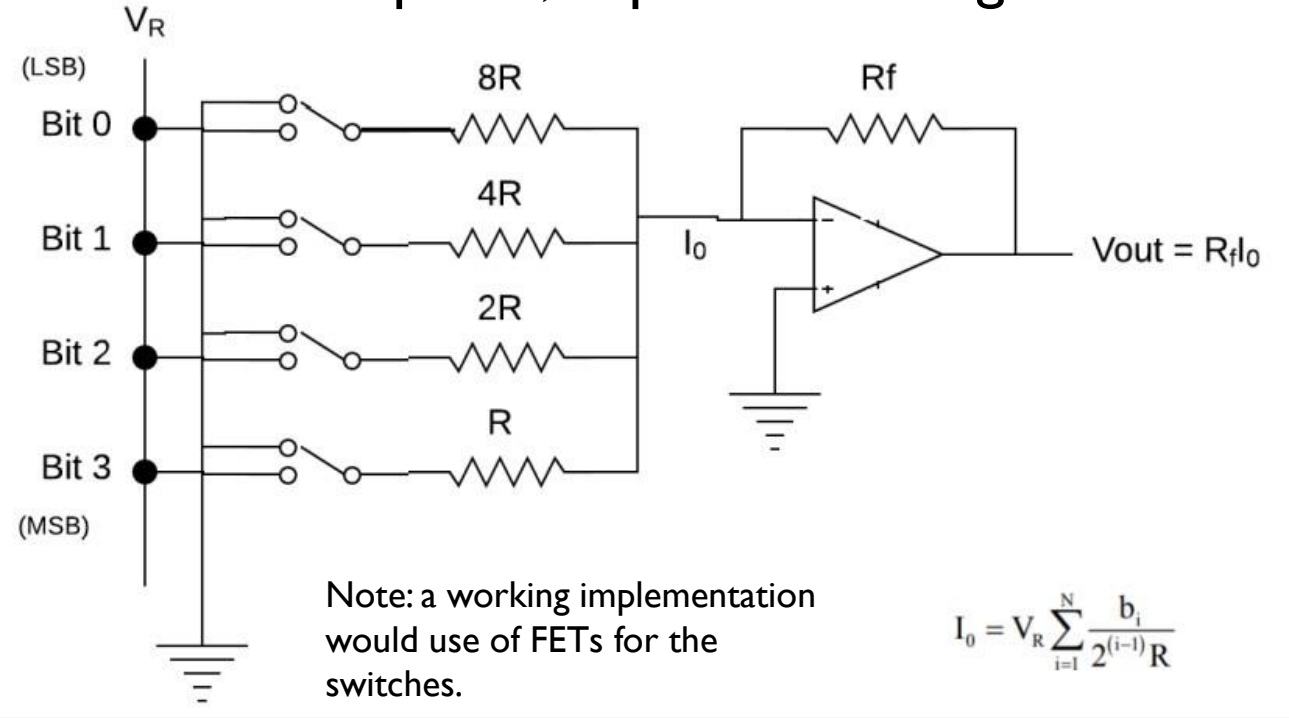
1001_2 → 5V / 16 (1*8 + 0*4 + 0*2 + 1*1) = 5*(8 + 1) / 16 = 2.8125V

R2R LADDER DAC: PROS & CONS

- Advantages:
  - Efficient
  - Uses only two resistor values: Well suited for silicon fabrication
  - Easily scaled to any N
  - Output impedance is R regardless of N
- Disadvantages
  - Instability, temporal fluctuations with use, eg. as some resistors heat and other's don't the conventions can become inconsistent (especially if some codes are used infrequently, then conversion of those codes may get quite erroneous)

BINARY WEIGHTED DAC (BW-DAC)

- The BW-DAC involves connecting a set of resistors to an op-amp's summing junction, generating an output that is proportional to a weight sum of the input voltages.
- *Advantages:* Simple to implement & fast conversion
- *Disadvantages:* Needs many high precision resistors; Low switch resistance required; Expensive for high bit numbers



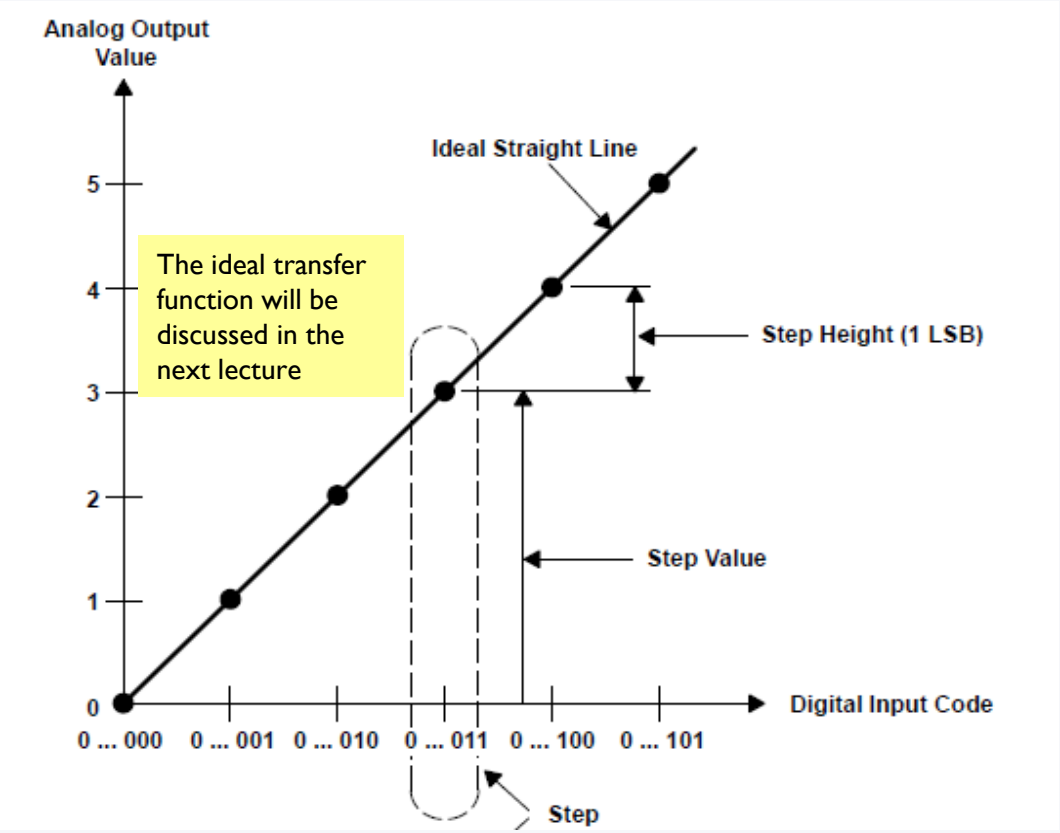
Note: a working implementation would use of FETs for the switches.

$$I_0 = V_R \sum_{i=1}^{N} \frac{b_i}{2^{(i-1)}R}$$

[5]

DAC METRICS

- You can apply most of the metrics we will discussed soon for ADCs to DACs



Ideal transfer function: used in showing accuracy of a DAC or ADC

Generally the difference (what you change for the ADC case) is that *h(t)* becomes the ideal transfer function, what the ADC should ideally output for the quantized inputs, and *w(t)* becomes the actual output.

(generally you want the DAC to have an ideal transfer function that is linear, but this isn't necessarily always the case)

DONE WITH DACS

- Generally, the basic design of a DAC is fast, efficient and not as demanding on resources as is the case for responsive ADCs.
- Therefore, I find it unnecessary to explore other DAC design (like switch capacitor network, chained DACs, etc.) for the purposes of this course.

Full up on knowledge of DAC
for likely embedded systems needs