Operationnal Amplifiers part 1 : bread & butter

May 17, 2019

Ideal operational amplifiers are a good start for calculating a signal chain. Today's mixed signal chips (sometimes called system on a chip SOC) hold dozenens of amplifiers. And sad but true: If you want to make money with your design the most important parameter is cost! As a consequence the most typical bread & butter OPAMP is far from ideal.

Of course as a top level designer you can always ask your module designers for higher performance, but that costs money, current, test effort, trimming, sophisticated circuits like auto zero or chopping.... So for ecconomical reasons better stick with bread & butter wherever it is possible and only use high performance amplifiers where unavoidable.

1 A bit of application theory

This first introduction is only intended to give a first draft overview of the use of an amplifier. We will need some of the equations later looking at the impact using our bread & butter OP.

1.1 Open loop operation

Open loop operation means the amplifier is working without feedback.

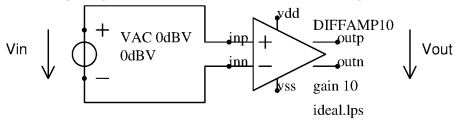

Fig.1.1.1: Open loop operation of an amplifier

This kind of operation of course only makes sense if the gain of the amplifier is well defined by design and stable over temperature.

$$V_{out} = V_{in} * gain_{amp} \tag{1}$$

If the input signal has a swing such that the output signal would go beyond the supply rails the amplifier simply starts clipping the signal. This becomes especially important if the amplifier has a very high gain (for instance 1000) and the input signal has an amplitude of several 10mV.

The most important advantage of open loop operation is that the amplifier can be designed for a high bandwidth without stability problems. Therefore open loop amplifiers are prefered for RF applications.

1.2 Closed loop operation

In closed loop operation of an amplifier a part of the output signal si fed back to the input. To keep things simple in the following we use an operational amplifier with single ended output.

Usually operational amplifiers have a very high gain (at least for low frequencies).

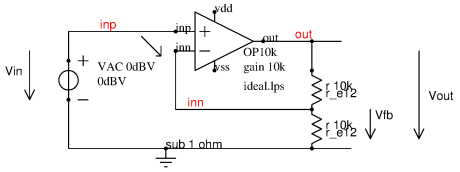

Fig.1.2.1: Amplifier in closed loop operation

We are interrested in the output signal Vout. Of cource Vout is the difference between V(inp) and V(inn) times the gain of the amplifier.

$$V_{diff} = V(inp) - V(inn) \tag{2}$$

$$V_{out} = gain_{amp} * V_{diff} \tag{3}$$

The voltage of node inn depends on the resistor divider and V_{out}

$$gain_{fb} = \frac{R_2}{R_1 + R_2} = \frac{1}{2}$$

$$V(inn) = V_{out} * gain_{fb}$$

$$V_{out} = gain_{amp} * (V_{in} - V_{out} * gain_{fb})$$

$$V_{out} = V_{in} * \frac{gain_{amp}}{1 + gain_{amp} * gain_{fb}} \tag{4}$$

Example: $gain_{amp} = 10000$, $gain_{fb} = 1/2$ lead to $V_{out}/V_{in} = 10000/(1+5000) = 1.9996$

The ratio

$$gain_{closed} = \frac{V_{out}}{V_{in}} \tag{5}$$

is called the closed loop gain. As long as the product $gain_{amp} * gain_{fb} >> 1$ the closed loop gain is close to $1/gain_{fb}$. An amplifier with unlimited gain would reach exactly this closed loop gain. The difference between the output signal of a

(theoretical) amplifier with unlimited gain and the real amplifier (with limited gain) is the error signal.

$$V_{error} = V_{out_{ideal}} - V_{out}$$

$$V_{error} = V_{in} * \frac{1}{gain_{fb} * (1 + gain_{amp} * gain_{fb})} \tag{6}$$

Often we are more interested in the relative error $V_{error}/V_{out_{ideal}}$

$$Err_{rel} = \frac{1}{1 + gain_{amp} * gain_{fb}} \tag{7}$$

Example: $gain_{amp} = 10000$, $gain_{fb} = 1/2$ lead to $Err_{rel} = 1/(1+10000*0.5) = 1.9996 * 10^{-4}$

2 The bread & butter OPAMP

The amplifier I call the bread & butter OPAMP is probably the most frequently used amplifier design of the analog world! I know mixed signal chips with more than 30 of them on the same chip. Typically this bread & butter OPAMP is used as:

- regulator amplifier (voltage regulators etc.)
- feedback amplifier in bandgaps and bias blocks
- unity gain buffer amplifier for on chip test analog test bus application
- and many more low performance applications

The main design target is low cost and low silicon real estate. Typical supply voltages range from 2.5V to 5V depending on the technology used. Here is an example using a 3.3V technology.

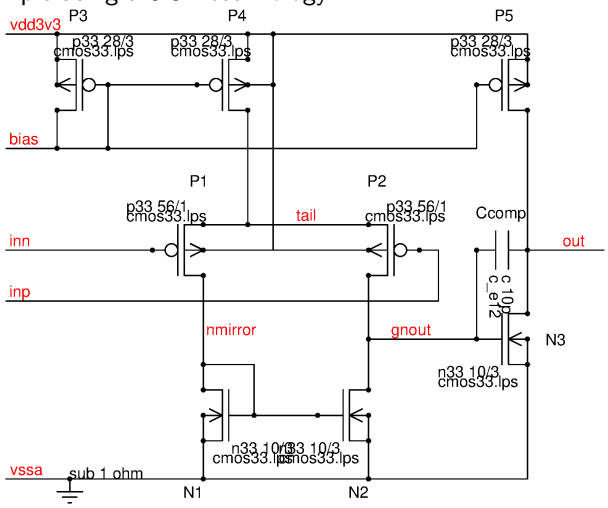

Fig.2.1: A very simple operational amplifier frequently used in mixed signal chips

Looking at the circuit we can directly see some of its limitations:

common mode range: The input stage can only be operated in a linear way in the following range:

$$V_{gsN1} + 4 * V_t - V_{gsP1} < V_{cm} < vdd3v3 - V_{gsP1} - V_{gsP4} + V_{thP4}$$

The left side of the equation determining the lower end of the common mode range assumes we need at least 4 times k*T/e as a drains-source voltage of the differential pair. This is about true if the differential stage operates close to weak inversion. The right side of the equation determines the upper limit of the common mode range. Since the current generators typically don't operate close to weak inversion here we really have to apply the gate overdrive of P4 ($V_{gsP4} - V_{thP4}$). Using a 3.3V technology a gate overdrive of about 0.5V is a usual engineering practice. Threshold voltage in a 3.3V process is typically 0.8V at room temperature. Including spread and cold temperature (-40 deg. C) the thresholds can reach up to 1.2V. Assuming the thresholds of the NMOS transistors and PMOS transistors match with an error of +-200mV the resulting common mode range becomes:

$$300mV < V_{cm} < vdd3v3 - 1.7V$$

This equation already shows why this simple circuit is not usable for supply voltages below 2.5V anymore. For lower supply voltages more sophisticated circuits with folded cascodes are required. Therefor current consumption starts to increase again for the analog parts using technologies with less than 2.5V supply voltage.

output drive: The output has a strong pull down N3 able to pull down several $100\mu A$. The pull up current is determined by the current generator P5. Typical bias currents are in the range of $1..50\mu A$. As long as the output just has to drive some CMOS gates or a high resistive feed back divider this isn't a problem. If low impedance loads have to be driven the circuit already needs some enhancements.

Maximum differential input voltage: The maximum permissible differential input voltage is limited by the dielectric strength of the gate oxide of P1 and P2. Up to about 5V the gate oxide dielectric strength normally follows the process voltage. Above 5V this is no more the case.

Slew rate: The slew rate is determined by the bias current and the frequency compensation capacitor Ccomp.

$$dV_{out}/dt = I_{bias}/Ccomp$$

Example:

with a bias current of $20\mu A$ and a frequency compensation of 10pF the slew rate becomes $2V/\mu s$.

Input offset: The statistical input offset mainly depends on the gate area of P1 and P2 and the matching parameters. As a rule of thumb (using SiO2 gate oxides) the matching is about:

$$Vos_{proc} = 1mV * t_{ox}/nm$$

The oxide thickness approximately scales with the process voltage.

$$t_{ox} = \frac{V_{gsmax}}{0.5V/nm}$$

Example:

A 3.3V process typically has a 7nm gate oxide. This leads to a matching constant of about $Vos_{3v3proc} = 7mV\mu m$.

The 1 sigma input offset voltage calculates:

$$V_{os1s} = \frac{Vos_{proc}}{\sqrt{W * L}}$$

Example:

The OPAMP shown has a 1s offset of about $V_{os1s} = 7mV\mu m/\sqrt{56\mu m * 1\mu m} = 0.935mV$. Since the manufacturer of the chip wants a good production yield and due to the number of amplifiers on one chip the specification usually states a 6 sigma value of 5.6mV.

To improve the offset, the input pair can be made bigger. Half the offset means 4 times the area and 4 times the current consumption to achieve the same speed again!

DC voltage gain: The DC voltage gain of the amplifier is the product of the gain from the input to node gnout (gain1) and from gnout to node out (gain2). Assuming the load impedance is significantly lower than the output impedance of N3 and P5 gain2 becomes:

$$gain2 = R_{load} * gm_{N3}$$

The transconductance of N3 can roughly be estimated from the aspect ratio W/L, the gate oxide thickness t_{ox}, electron mobility μ_{esi} of silicon and the bias current I_{bias}. (The equation is just the result of solving the classical MOS transistor equations for strong inversion)

$$gm_{N3} = \frac{dI_d}{dV_{gs}} = 2 * \sqrt{\frac{W}{L} * I_{bias} * \frac{\mu_{esi} * \varepsilon_{sio2}}{2 * n * t_{ox}}}$$

The factor n is the gate coupling factor. for most technologies it is in the range of 1.2..1.6. In the following let's assume n=1.4. The dielectric constant of silicon oxide is $\varepsilon_{sio2} = 0.34pAs/Vcm$. Electron mobility of silicon is about $\mu_{esi} = 600cm^2/Vs$.

Example:

Using our example of a 3.3V process and 7nm gate oxide and a bias current of $20\mu A$we get

$$gm_{N3} = 2 * \sqrt{\frac{10}{3} * 20\mu A * \frac{600cm^2 * 0.34 * 10^{-12}As}{Vs * Vcm * 2 * 1.4 * 7 * 10^{-7}cm}} = 385\frac{uA}{V}$$

Assuming we have to drive a load of $100K\Omega$ the voltage gain of the output transistor becomes 38.5 .

In a similar way we can calculate gain1. First we need the DC-impedance of node gnout. For DC the impedance is determined by the early voltage of the shortest transistor connected to this node. Normally (like in the example) this is P2. The DC impedance depends on the early voltage. As a rough estimation the early voltage is about:

$$V_{early} = L * 10V/\mu m$$

The resulting DC resistance becomes:

$$R_{gnout} = V_{early}/I_{P2} = 2 * V_{early}/I_{bias}$$

In out example we get $R_{gnout} = 10V/10\mu A = 1M\Omega$.

To calculate the gain we have to use the same equation as before with two modifications:

1. We have to use the mobility of holes in stead of electrons $\mu_{hsi} = 200cm^2/Vs$
2. We have to use the W and L of P1 and P2

$$gain1 = R_{gnout} * 2 * \sqrt{\frac{W_{P1P2}}{L_{P1P2}} * I_{bias} * \frac{\mu_{hsi} * \epsilon_{sio2}}{2 * n * t_{ox}}}$$

Example:

$gain1 = 10^6\Omega * 2 * \sqrt{\frac{56}{1} * 20\mu A * \frac{200cm^2 * 0.34 * 10^{-12}As}{Vs * Vcm * 2 * 1.4 * 7 * 10^{-5}cm}} = 39.42$

The total gain of out bread & butter amplifier eventually is

$$gain = gain1 * R_{load} * gm_{n3} = 38.5 * 39.4 = 1516$$

or 64dB.

Not much compared to a high performance OPAMP. But does it really harm? Let's operate this bread & butter OPAMP in a closed loop and sweep the target gain of the closed loop from 1 to 1000 and observe the relative error. The calculation just uses the error calculation of the closed loop (7).

$$Err_{rel} = \frac{1}{1 + gain_{amp} * gain_{fb}}$$

with $gain_{fb} = 1/gain_{ideal}$

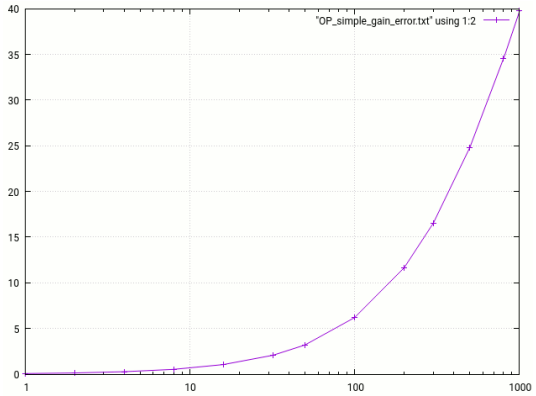

Fig.2.2: Gain error of the closed loop in % sweeping the closed loop target gain from 1 to 1000

The error versus target closed loop gain shows that as long as the closed loop gain is kept in the range of 1..10 the error is in the range of 1% or less. Using the bread & butter OPAMP can be justified. Using such a low performance OPAMP for higher closed loop gains will lead to problems.

Gain bandwidth product: The calculation of the gain applies to higher frequencies as well. The gain of the first stage has to drive the miller capacity Ccomp.

$$gain(f) = gm_{P1P2}/j\omega C_{comp}$$

$$gain(f) = \frac{1}{j * \omega * C_{comp}} * 2 * \sqrt{\frac{W_{P1P2}}{L_{P1P2}} * I_{bias} * \frac{\mu_{hsi} * \epsilon_{sio2}}{2 * n * t_{ox}}}$$

The unity gain frequency or gain-bandwidth-product is

$$f_{unity} = \frac{1}{2 * \pi * C_{comp}} * 2 * \sqrt{\frac{W_{P1P2}}{L_{P1P2}} * I_{bias} * \frac{\mu_{hsi} * \epsilon_{sio2}}{2 * n * t_{ox}}}$$

Example:

In our example this is $f_{unity} = 314kHz$.

Not really a race horse, but for standard applications such as a unity gain buffers for slow signals or even DC this simple amplifier is good enough and - from sales point of view much more important - it is small and cheap. Of course, there are much better designs. These should only be used where the high performance is needed.

3 Conclusion

It is possible to design a fairly cheap OPAMP as long as the requirements don't get too high. Certain restrictions regarding common mode range, output drive

capability, offset and gain-bandwidth product however have to be accepted. But it is worth while trying to bring down the design cost.

Specifying parameters that go beyond the real requirements drive chip cost as well as design time, test effort and current consumption. Only specify difficult to achieve parameters where you really need it!