

Microelectronics Essentials

Philipp Häfliger

8th May 2026

Edition 1.8

Department of Informatics, University of Oslo, Oslo 21-Jan 2025
E-book ISBN (Edition 1.0): 978-82-7368-467-7

Copyright © 2025 Philipp Dominik Häfliger

This work is licensed under a Creative Commons
“Attribution-NonCommercial-NoDerivatives 4.0 In-
ternational” licence.



Change Log

edition	date	change
0.1 (beta)	22-Jan-2024	base beta edition, ISBN: 978-82-7368-466-0
0.15 (beta)	31-May-2024	last beta edition update
1.0	20-Dec-2024	first edition, ISBN: 978-82-7368-467-7
1.1	21-Jan-2025	added relation U_T to temperature in chapter 2. Added minus sign to $\frac{1}{\lambda}$ in caption of Fig. 2.1, deep conduction def in chapter 2 amended to $V_{DS} \ll V_{sat}$. Added appendix C with some Matlab code.
1.2	10-Feb-2025	clarification in the task text for A.1.11 and a small correction in the text for solution A.1.11: '.... $V_D < V_{OVn} = 1.35V$...'
1.3	7-Apr-2025	Added deduction of LF small signal G_M, R_O, R_I for CG stage in appendix B.2.1
1.4	5-May-2025	Limitation of cascode current mirror for max current added (10.37). In caption of figure 7.8 and in the text nearby clarified about increasing W and leaving L constant.
1.5	7-July-2025	Some words on biasing the folded cascode gain stage in caption to fig. 9.3 and differential folded cascode trans-amp in section 11.4.1. Exercises, task A.4.6: corrected reference from chapter 5 to section 9.4.1. Line break for solution A.1.7. Adding 2025 exam tasks to exercises.
1.6	5-Mar-2026	Some large signal model edits and improved figures. Corrected solution for A.2.4. Correction in (3.7)
1.7	16-Apr-2026	Condition for general transconductance amp simplifications made precise in sections 9.4.3 and 9.4.4. Exercise A.4.1 made more precise with respect $I_D = 40\mu A$ and I'_D . Specified figure reference in A.4.5. Mentioning Vdd as a process parameter in section 2.1. Mentioning in 9.2.1 that saturation is usually expected at the POO for amplifiers to work as intended.
1.8	8-May-2026	Corrected in CS-discussion on intrinsic gain and small signal model section expressions for g_m and r_{ds} for I'_D vs I_D , i.e. equations (9.9) and (3.7). Added index for g_m and some others. Correction in equation (11.8): added $2g_m$ on third line. Corrected/modified some model RO conditions to accommodate the simplifying assumption the $ I_F > I_R $. In exercise A.4.10 added $C_L = 1pF$ that was missing.

Abstract

While research is on the hunt for an even smaller replacement, silicon as a base material and the microscopic field effect transistor (FET) is for now still the essential building block of microelectronics. A central- or a graphics processing unit (CPU/GPU) contains billions of the things, and sensors and actuators in our everyday commodity devices, industrial machines, medicine and space exploration, are read-out and controlled by transistor circuits.

This lecture script aims at enabling the reader to use FET models to construct some of the more common composite building blocks of both digital and analog microelectronics. It teaches how to manipulate design parameters at transistor level to optimize logic gates and simple amplifiers, and showcases the inner workings of those devices. The reader shall learn their non-idealities and pitfalls to better be able to use them with confidence for higher order digital-, analog-, and mixed-signal systems, and to tweak the elemental electronics building blocks for optimal performance.

Acknowledgment

This book results from a lecture script for a basic microelectronics course at the University of Oslo and was first used in lieu of pre-existing standard textbooks in the spring term of 2024. Consecutive editions have been improved based on the feedback of the students in the course and by colleagues at the department. I want to shout out to them and give them my heartfelt thanks!

Contents

Abstract	v
Acknowledgment	vii
Prologue	xxi
I Transistor Models	1
1 Field Effect Transistor Cross Sections	3
2 FET Large Signal Models	9
2.1 I-V relationships	9
2.2 Capacitances	19
3 FET Small Signal Models	21
3.1 FET behaviour linear approximation around a point of operation	21
3.2 Small signal equivalent circuit	24
3.3 FET low frequency small signal model	27
3.4 Applying small signal analysis	28
3.5 FET high frequency small signal models	32
4 FET Switch Models	35
4.1 Static output resistance R_{ON}	39
4.2 Dynamic transitions and propagation delay	39
4.2.1 Approximating a constant I_{ON} for signal transition . . .	39
4.2.2 Approximating a constant R_{ON} for signal transition . . .	42
4.2.3 What's next	43
II Linear Circuit Analysis Basics	45
5 Linear circuit filters	47
5.1 Linear filter and transfer function	47
5.2 Deriving a transfer function from a linear circuit	54
5.3 Thévenin and Norton equivalent circuit, input- and output im- pedance	55
5.4 Bode plot, poles and zeros	56
5.4.1 Root form of the transfer function	58

5.4.2	How to draw Bode plots from the real poles and zeros of a transfer function	61
5.5	Steps for deriving the transfer function and its Bode plot of a linear circuit	62
5.6	Dominant pole approximation	62
5.7	Stability	63
5.7.1	2nd order equation with complex poles	64
5.7.2	Negative feedback, loop gain, and phase margin	71
5.8	Approximating higher order with 1 st order or 2 nd order	79
III	Digital Circuits Basics	81
6	Boolean Logic	83
7	Combinational Logic Circuits	85
7.1	Complementary PU/PD Logic	85
7.1.1	Balancing PU and PD networks	89
7.1.2	Inverter	90
7.1.3	Inverter: Miller approximation of C_O for step input	91
7.1.4	More complex gates	92
7.2	Dynamic and static power consumption	94
7.3	Driving large external loads C_W	96
7.4	Dynamic logic	97
7.5	Example of advanced combinational logic circuits	103
8	Sequential Logic Circuits	105
8.1	Asynchronous latches	105
8.2	Synchronous flip-flops	108
8.3	Finite state machines	108
8.4	Example list of more advanced sequential logic circuits	113
IV	Analog Circuit Basics	115
9	Single Ended Amplifiers	117
9.1	General transconductance amplifier low frequency/DC analysis .	117
9.2	Single transistor amplifier low frequency/DC analysis	120
9.2.1	Biasing/point of operation	120
9.2.2	Common Source (CS) gain stage	120
9.2.3	CS/transistor intrinsic gain	121
9.2.4	The Common Gate (CG) gain stage	122
9.2.5	The Common Drain (CD) stage or Source Follower (SF) .	122
9.3	The Cascode gain stage	124
9.4	General transconductance amplifier frequency response	125
9.4.1	Exact solution	125
9.4.2	Without feedback capacitor	127
9.4.3	Assuming dominant pole at output	128
9.4.4	Assuming dominant pole at input	128
9.4.5	Miller approximation assuming dominant pole at input . .	129

9.5	Single transistor gain stages frequency response	130
9.5.1	Intrinsic CS/transistor speed	131
9.5.2	Trade off of intrinsic speed vs gain	132
9.6	Stability of single transistor stages	132
9.7	Cascode gain stage frequency behaviour	135
10	Current Mirrors	137
10.1	Simple CMOS implementation	138
10.1.1	Large signal output current offset	139
10.1.2	Range of operation	139
10.1.3	Small signal LF/DC gain	140
10.1.4	Frequency response	141
10.2	Cascode output	145
10.3	Cascode at output and input	145
10.4	Automatic biasing of output cascode	149
10.4.1	Double diode input, cascode output	149
10.4.2	The Wilson CM	150
10.4.3	The modified Wilson CM	151
11	Differential Amplifiers	153
11.1	Differential pair	153
11.2	R-load gain	156
11.3	Differential transconductance amplifier	158
11.3.1	Differential gain	158
11.3.2	G_M and R_O deduced separately	161
11.3.3	Common mode rejection ratio (CMRR)	163
11.3.4	Systematic offset	165
11.3.5	Frequency response	166
11.3.6	Slew rate	169
11.4	Higher gain operational differential amplifiers introduction	171
11.4.1	Folded cascode operational transconductance amplifier	171
11.4.2	Two stage operational transconductance amplifier	173
	Index	175
	Bibliography	178
V	Appendices	181
A	Exercises	183
A.1	Transistor Models	184
A.1.1		184
A.1.2		184
A.1.3		184
A.1.4		184
A.1.5		185
A.1.6		185

	A.1.7	186
	A.1.8	186
	A.1.9	187
	A.1.10	187
	A.1.11	188
	A.1.12	188
	A.1.13	188
A.2	Linear Circuit Analysis Basics	196
	A.2.1	196
	A.2.2	196
	A.2.3	196
	A.2.4	198
	A.2.5	198
	A.2.6	198
	A.2.7	201
A.3	Digital Circuit Basics	206
	A.3.1	206
	A.3.2	206
	A.3.3	207
	A.3.4	207
	A.3.5	207
	A.3.6	208
	A.3.7	208
	A.3.8	209
	A.3.9	209
	A.3.10	212
	A.3.11	212
	A.3.12	213
	A.3.13	213
	A.3.14	215
	A.3.15	216
	A.3.16	217
	A.3.17	218
A.4	Analog Circuit Basics	226
	A.4.1	226
	A.4.2	227
	A.4.3	227
	A.4.4	227
	A.4.5	228
	A.4.6	229
	A.4.7	230
	A.4.8	230
	A.4.9	232
	A.4.10	232
	A.4.11	232
	A.4.12	232
	A.4.13	233
	A.4.14	233
	A.4.15	234
	A.4.16	235

A.4.17	235
B Proofs and deductions	251
B.1 Linear Circuit Analysis Basics	252
B.1.1 Stability	252
B.2 Analog Circuit Basics	253
B.2.1 Single transistor amplifier LF/DC analysis	253
C Matlab code	257
C.1 The full EKV model	258

List of Figures

1.1	Bulk CMOS nFET and pFET cross section	4
1.2	SOI FET cross section	4
1.3	FinFET cross section	5
1.4	FET Symbols	6
2.1	How to derive λ graphically.	12
2.2	Typical I_D vs V_{DS} plots for varying λ	13
2.3	Typical I_D vs V_{GS} and V_{DS} plots for varying I_S	13
2.4	Typical I_D vs V_{GS} and V_{DS} plots for varying V_t	14
2.5	Four regions of operation where the simplified FET large models can be applied	14
2.6	Illustrating the shortcomings of the simplified models in specific regions of operation	15
3.1	Linearised FET model	22
3.2	Linearised FET schematics	23
3.3	Linearised electronic system schematics	25
3.4	Low frequency FET small signal equivalent circuit	26
3.5	Low frequency FET small signal equivalent circuit variant	26
3.6	CS amplifier circuit and its small signal equivalent circuit.	29
3.7	CS stage modelling	29
3.8	High frequency FET small signal equivalent circuit	32
4.1	General purpose switch possibilities for both analogue and digital	36
4.2	FET switch models	37
4.3	Comparing large signal FET model with switch models	38
4.4	Models for transition time/propagation delay	41
5.1	Passive linear circuit elements	48
5.2	Linear combination of sine wave signals	50
5.3	Linear combination of sine waves at $t = 0$	51
5.4	Ideal active linear circuit elements	53
5.6	Thévenin and Norton equivalent circuits	56
5.7	Example Bode plot	57
5.8	Contribution of a single real negative pole to the Bode plot	59
5.9	Contribution of a single <i>negative</i> zero to the Bode plot	60
5.10	Contribution of a single <i>positive</i> zero to the Bode plot	61
5.11	2nd order low pass with $Q = 0.1$ and $\omega_0 = 10^4 \text{Rad}$	64

5.12	2nd order low pass with $Q = 0.5$ and $\omega_0 = 10^4 \text{Rad}$	65
5.13	Complex plain depiction of 2nd order low pass denominator growth when $Q = \frac{1}{\sqrt{2}}$	67
5.15	2nd order low pass with $Q = 2$ and $\omega_0 = 10^4 \text{Rad}$	69
5.16	General concept of a feedback loop	71
5.17	Feedback amplifier circuits for specific closed loop gains	72
5.18	Adapting feedback β to improve phase margin	74
5.19	Pole separation to improve phase margin	75
7.1	Boolean operators, truth tables and logic gates	86
7.3	Ways of implementing a NAND gate.	88
7.4	Boolean operands that are most easily implemented as static logic circuits	89
7.5	Miller effect inverter	91
7.6	Balancing logic gates	93
7.7	Illustration of inverter cross current	95
7.8	Cascade of ever bigger inverters driving big load	96
7.9	Generic dynamic logic implementation of a digital multiplexer	99
7.10	Domino logic implementation of a digital multiplexer	101
7.11	NP domino logic implementation of a digital multiplexer	102
7.12	Dual rail domino logic for asynchronous circuits	103
8.1	Double inverter feedback loop and D-latch aka transparent latch	106
8.2	RS-latch	106
8.3	D-flipflop implemented with master slave D-latch pair	107
8.4	D-latch state transition graph	108
8.5	Asynchronous D-flipflop state transition graph	109
8.6	Synchronous D-flipflop state transition graph	110
8.7	State transition graph of a traffic light controller	111
8.8	General structure of a finite state machine (FSM)	111
8.9	Schematic of the traffic light final state machine	112
9.1	General low frequency model of a (transconductance) amplifier	118
9.2	common gate, source, and drain stages large- and small signal schematics	119
9.3	Cascode gain stage	123
9.4	General transconductance amplifier model for frequency analysis	125
9.7	Miller approximation of inverting amplifier input capacitance	129
9.8	Cascode gain stage frequency equivalent model as two single transamp stages in sequence	135
9.9	Cascode gain stage frequency model assuming dominant pole at input	135
9.10	Cascode gain stage frequency model assuming dominant pole at output	136
10.1	Current mirror symbol	137
10.2	A current mirror's simplest two transistor CMOS implementation	138
10.3	A general single stage current amplifier model	141
10.4	Current mirror with cascode output	144
10.5	Non-functional cascode I/O current mirror	146

10.8 Wilson Current mirror	150
10.9 Modified Wilson CM	152
11.1 Differential amplifier symbols	153
11.2 diff pair large signal	154
11.3 Small signal diff pair diff I output	154
11.4 Small signal diff pair diff I output	154
11.5 diff pair large signal V out	157
11.6 Small signal diff pair V output	157
11.13 Voltage follower	169
11.14 Slew rate illustration	170
11.15 Folded cascode operational transconductance amplifier	172
11.16 Two stage differential amplifier	174
A.1 Exercise: compute λ from I_D vs V_{DS} graph	184
A.2 Exercise: compute r_{ds} near origin from I_D vs V_{DS} graph	185
A.3 Schematic for small signal analysis	186
A.4 Balance between a fully conducting pFET and nFET pulling respectively up to Vdd and down to Gnd in a 180nm process with Vdd=1.8V	192
A.5 Template to draw Bode plot	197
A.6 Bode plots for task A.2.2	198
A.7 Bodeplot for transfer function H1 for task A.2.5	199
A.8 Bodeplot for transfer function H2 for task A.2.5	200
A.9 Bodeplot for transfer function for task A.2.7	201
A.10 Solution A.2.1	202
A.11 PDN and corresponding PUN candidates	207
A.12 Digital gates for task A.3.5	208
A.13 Digital gates for task A.3.5	209
A.14 Digital gates for task A.3.7	210
A.15 Combinational logic circuit for task A.3.8	211
A.16 Logic to be implemented as NP domino in task A.3.10	212
A.17 Symbols to use in task A.3.11	212
A.18 Representations of the 25 states of the FSM described in task A.3.14	214
A.19 Multi-inverter digital buffer illustration for task A.3.15	216
A.20 Gates to be implemented as NP Domino logic in task A.3.16	217
A.21 Solution task A.3.3	219
A.22 Solution task A.3.9	221
A.23 NP domino logic solution for A.3.10	222
A.24 Solution task A.3.12	222
A.25 Exponential Decay counter state transition graph solving A.3.13	223
A.26 Solution task A.3.14	224
A.27 Solution task A.3.16	225
A.28 Solution task A.3.17	225
A.29 CS stages to be ordered by their gain with same input bias V_I	227
A.30 Generic CS stage for task A.4.4	228
A.31 Figure for task A.4.5: Current conveyor	229
A.32 Source follower for task A.4.6	230

A.33 Schematic of a current mirror loaded differential pair with ideal bias current source	231
A.34 CS small signal schematics to compute Miller capacitance	231
A.35 Double Cascode Current Mirror	234
A.36 Equivalent schematic for input impedance.	235
A.37 Measurement to deduce input impedance.	236
A.38 Measurement to deduce CMRR	237
A.39 Illustration for the solution of task A.4.13	245
A.40 Illustration for the solution for task A.4.14	247

List of Tables

5.1	Passive impedances Ohm's law	52
5.2	A table of ideal active circuit elements we'll encounter in this compendium in linear electronic systems.	53
5.3	Relation PM, Q, and ratio of 2nd pole to unity gain	79
6.1	Symbols used for Boolean operands	83
6.2	Basic Boolean functions truth tables	84
6.3	Rules for Boolean arithmetic	84
8.1	D-latch characteristic table	106
8.2	NOR gate implementation of RS-latch characteristic table	107
8.3	Characteristic table traffic light controller	112
9.1	Equivalent linear amplifier parameters for single transistor amplifier configurations.	121
9.2	Single transistor gain stages capacitors assigned to the general transconductance amplifier model.	125
9.3	Single transistor amplifiers time constants when the input node dominates	130
9.4	Single transistor amplifiers time constants when the output node dominates	130
10.1	Equivalent linear amplifier parameters for CS and CM.	142
A.1	Process parameter examples	226

Prologue

This is a book about CMOS integrated circuits. CMOS technology with the field effect transistor (FET) at its heart is (and will remain so for some time still) the predominant technology upon which our electronics is built.

The cynical character Dogbert in a Dilbert cartoon once suggested that a standard component for any arbitrary presentation is a chart of Moore's law. I'll partially follow his advice here by at least remarking upon that law: it has quite precisely predicted the exponential growth in density of transistors from when they were first demonstrated as integrated circuits in 1958/1959 until today. Moore predicted in 1965 that the density would approximately double every year and revised his prediction in 1975 to a doubling every second year. Since then, his prediction has proven very apt. Some claim it has been a self-fulfilling prophecy as it has been adapted as a goal by the industry.

In recent years many experts have started to predict the end of that trend due to various physical limits to miniaturization. Interestingly, more recent voices attribute the coming end of Moore's law not so much to physical limits but rather to economical considerations: ever more companies that have been driving the miniaturization of integrated circuits have dropped out of the race, simply because their business model suggests they will never get their investment back. That is because not only the transistor density grows exponentially but also the development cost to achieve the next step in miniaturization. So only a handful of the biggest CMOS foundries remained and offer 7nm and smaller integrated circuit technology, while others do no longer drive towards more density and miniaturization, but offer more specialized properties of their integrated circuit devices and structures for specific niche markets.

In the meantime, completely different materials and computational principles start to enter into competition with (or supplement) semiconductor based integrated electronics, such as carbon nanotube transistors and quantum computers to name but two.

So while Moore's law will come to an end at some point in the near future, and FETs as the essential building block of computational devices are getting competition: for now and some time to come semiconductor electronics will still be the substrate on which most of our information processing will run, and the content of this book will remain useful for a while longer.

Philipp Häfliger
Oslo, 20-Dec-2023

Part I

Transistor Models

Chapter 1

Field Effect Transistor Cross Sections

Figures 1.1, 1.2 and 1.3 show cross sections of **complementary metal oxide semiconductor (CMOS) n-channel field effect transistors (nFETs)**, with a p-channel FET (pFET) mixed in in figure 1.1. CMOS FETs are also referred to as **MOSFETs**. They are four-terminal devices where the terminals are denominated as '**bulk**' (**B**), '**gate**' (**G**), '**source**' (**S**) and '**drain**' (**D**). nFETs have a (P-)-doped bulk and (N+)-doped source/drain, and vice versa for **pFET** devices. The '+' stands for a relatively high doping concentration and the '-' for a relatively low doping concentration. Oftentimes the device's anatomy is symmetric with respect to source and drain. Which of the two terminals is called what is actually determined by the voltages applied: for nFETs the 'source' is the terminal with the lower voltage and vice versa for pFETs. One of the most common uses of FETs is to use the voltage of the gate to control the conductance and/or current between the source- and drain terminals. Note that the gate terminal is insulated from the others. So while it can have different voltages, no DC current flows in or out. Pure electrostatic properties let its voltage modulate the conductive channel between source and drain by attracting or driving away free (i.e. mobile) electric carriers. In classical mode of operation the bulk is simply at a fixed voltage, $-V_{ss}/Gnd$ (i.e. the low supply voltage of an entire region of the integrated circuit) for P-bulks (i.e. nFETs) and V_{dd} (i.e. the high supply voltage for a region) for N-bulks (pFETs), but one actually *can* modulate its voltage and thus modulate the channel conductivity in the same manner as with the gate but with less efficiency. The bulk is then referred to as *back gate*. In **bulk CMOS** (figure 1.1) the P-bulk cannot be at a higher potential than the source/drain terminals, because then the PN-junction between bulk and the source/drain would be conducting and source/drain would be shorted with the bulk. So the range of 'allowed' voltages is more limited for the bulk than for the gate. This however, is not the case for a **silicon on insulator (SiO)** FET device (figure 1.2), where the back gate is more worry-free to use. Furthermore, SiO technologies have no PN-junction capacitance at their source/drain terminals, and thus less 'parasitic' capacitance that hamper the rapid change of voltages. Consequently, SiO technologies are popular for high speed and RF applications. The most modern CMOS technologies with gate

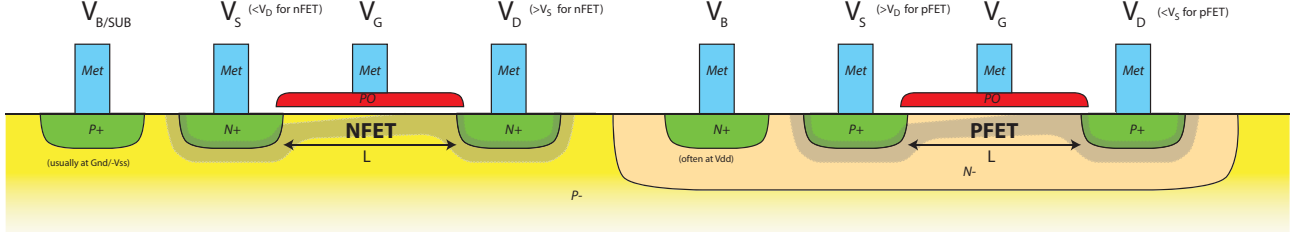


Figure 1.1: **Bulk CMOS** nFET (left) and pFET (right) cross section. Bulk CMOS is the original CMOS technology and still in use today, albeit not for the most downscaled CMOS processes. In normal biasing conditions, depletion regions (i.e. regions with no free/mobile carriers marked in grey) will insulate source and drain from the bulk, and separate a 'channel' of free carriers from the bulk as well. The fact that there is a depletion region in the PN-junctions is referred to as a 'reversed biased' PN-junction, i.e. with a lower voltage at the P-anode and a higher voltage at the N-cathode. The depletion region underneath the gate is due to the gate bulk voltage being biased in **inversion**, i.e. for the P-bulk a higher voltage on the gate than on the bulk, driving the positively charged majority carriers away from the surface and even inverting the channel by attracting minority carriers, i.e. freeing electrons from their bonds rendering them mobile. The opposite effect is known as **accumulation**, i.e. a lower voltage on the gate than the bulk that increases the number of free majority carriers. However, this is not a state a FET is biased in when controlling current flow between drain and source.

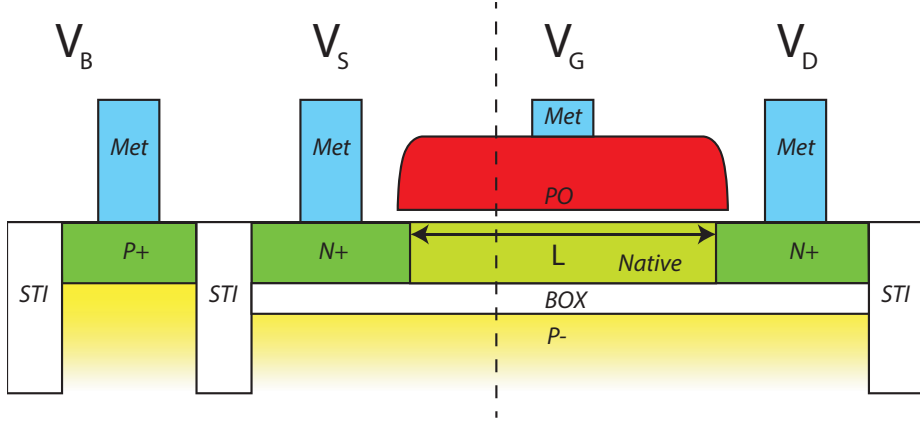


Figure 1.2: **Silicon on insulator (SOI)** nFET cross section for a variant with buried oxide (BOX). There are some other 'flavours' around, e.g. silicon on sapphire, where the oxide is not buried into the wafer substrate but silicon is deposited on an insulator substrate. Advantages compared to bulk CMOS are reduced capacitance at the source and drain terminals that do not have junction capacitance, as well as a bigger voltage range when using the bulk as back gate (No need to be wary of forward biased junction). In essence the cross section of a FinFET device looks the same along the channel, but is different in the cross section *across* the channel, i.e. along the dashed line. See figure 1.3

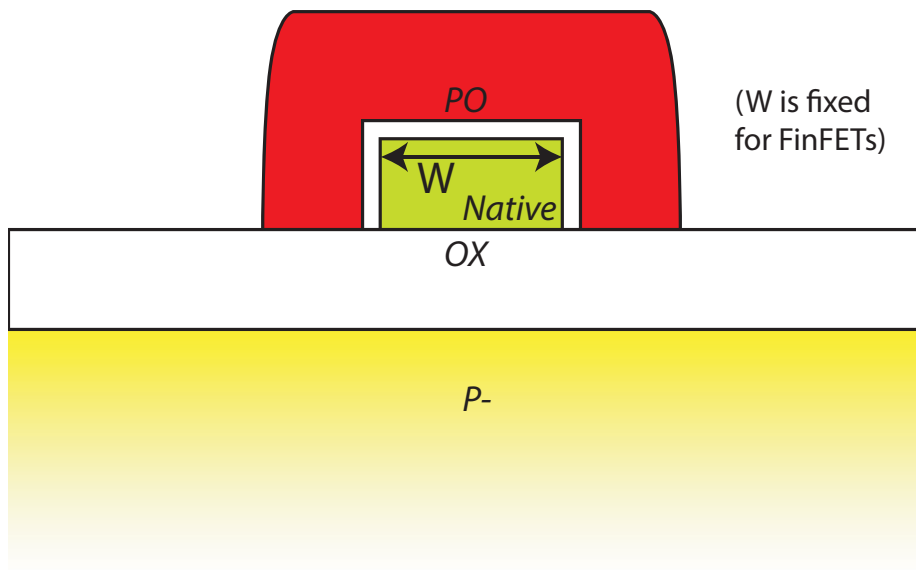


Figure 1.3: **FinFET technology** cross section *across* the channel. In contrast to Bulk and traditional SiO, here the gate 'folds' around the channel. FinFET technology is used for technologies of 14nm and smaller. A notable difference here is that the width of the channel is now fixed. So one has to use multiple transistors in parallel instead of just widening a channel. FinFETs can be produced both in SOI (shown here) and bulk CMOS (where there would be no (buried) oxide (BOX/OX) and the channel/bulk material might be doped instead of native).

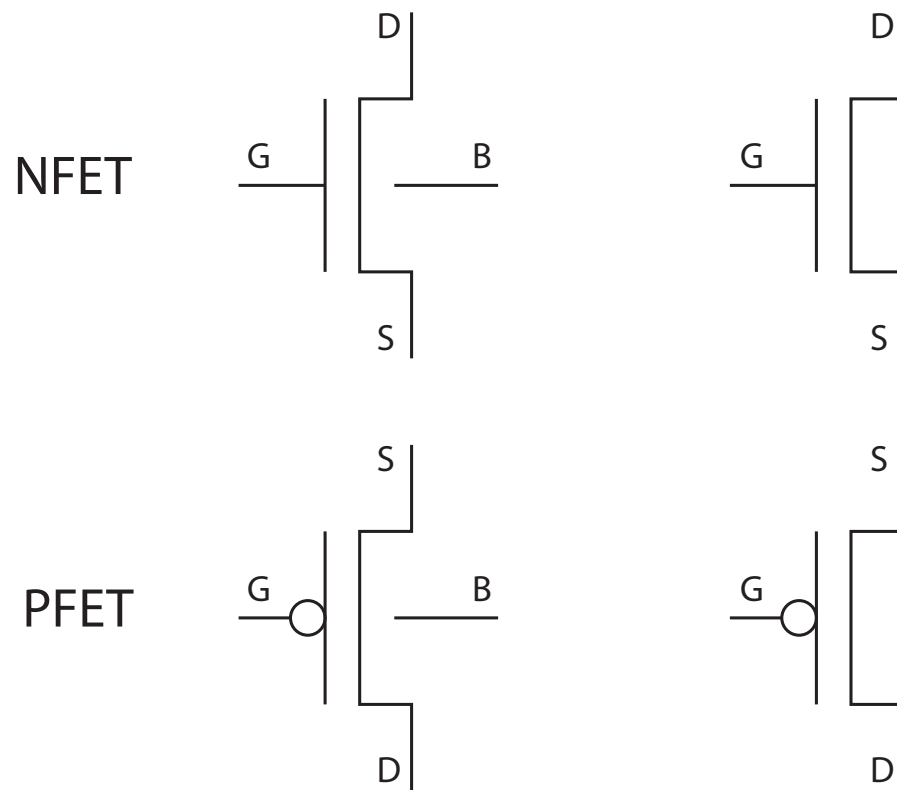


Figure 1.4: The symbols for NFETs and PFETs used in this book. If the bulk is omitted it is implicitly connected to $-V_{ss}/Gnd$ for nFETs and to V_{dd} for pFETs.

lengths below 14nm use **FinFET** topologies to ensure better control of channel conductivity at lower voltages for these maximally miniaturized devices. The gate is wrapped around the channel on three sides (see figure 1.3).

A set of commonly used symbols for nFETs and pFETs that will be used in this text is depicted in figure 1.4.

Chapter 2

FET Large Signal Models

2.1 I-V relationships

Large signal models are the most complete FET models discussed in this book and describe the transistor behaviour for any combination of a large range of voltages or currents applied to the four terminals of the MOSFET. The voltage range is limited by the **supply rails** given for a specific CMOS processing technology. Supply rails in this book shall be denominated as V_{dd} and $-V_{ss}$. Generally the signal zero reference $Gnd := 0V$ for (analog) circuits is between those rails. However, sometimes it is defined as $Gnd = -V_{ss} = 0V$, mostly for discussion of digital circuits.

For FETs the large signal model generally describes a function for the drain current I_D (the current into the drain terminal, and to a first approximation the same as exiting the source terminal) in dependency of the four terminal voltages V_S, V_G, V_D, V_B .

$$I_D = f(V_G, V_S, V_D, V_B) \quad (2.1)$$

The full model is often simplified some more, for instance by assuming that every voltage is referenced to $V_B := 0V$, oftentimes without explicitly mentioning this, so reducing the transistor to a three terminal device:

$$I_D = f(V_G - V_B, V_S - V_B, V_D - V_B) := f(V_{GB}, V_{SB}, V_{DB}) := f(V_G, V_S, V_D) \quad (2.2)$$

There are a few different schools of models. Here we shall base our discussion on the so called **EKV model** [3]:

$$I_D = I_F - I_R \quad (2.3)$$

Where the drain current I_D is the current *into* the drain node. The forward current I_F in the direction of I_D from drain to source and reverse current I_R in the opposite direction for nFETs are defined as:

$$I_{F(R)} = I_S \ln^2 \left[1 + e^{\frac{[V_G - V_{S(D)}] + (n-1)[V_B - V_{S(D)}] - V_t}{2nU_T}} \right] (1 + \lambda |V_D - V_S|) \quad (2.4)$$

Where the **characteristic current** I_S is

$$I_S = 2nkU_T^2 \quad (2.5)$$

and

$$k = \mu C_{ox} \frac{W}{L} \quad (2.6)$$

Most parameters in this equations belong into two categories, **process parameters** and **design parameters**.

Process parameters are given by the choice of CMOS technology to produce an integrated circuit and are usually provided by a CMOS foundry to its customers. They are the **threshold voltage** V_t , the **carrier mobility** μ , the **gate oxide capacitance per area** C_{ox} , the **slope factor** n ($1 \leq n < 2$), as well as the parameter λ' to derive the **channel length modulation parameter** λ that is $\lambda' = \lambda L = \frac{1}{V_A}$ (see further along). Also the nominal supply voltage V_{dd} - V_{ss} , i.e the standard voltage range limit for all the transistor's voltage terminals is a process parameter. Sometimes the foundry provides a small number of different device options in the same CMOS process, i.e. a set of device parameters that allows to produce transistors with different base properties in the same technology, e.g. very commonly there are different parameters for pFETs and nFETs (here we will distinguish these parameters with an extra index n or p , where relevant, such as V_{tn} or μ_p), but also different device options within these two major groups, such as low and high threshold devices with their own threshold voltages.

Then the design parameters are parameters that the circuit designer can arbitrarily choose within a given CMOS technology by drawing the device layout, i.e. the device's **channel width** W and **channel length** L .

Finally the **thermal voltage** $U_T := \frac{kT}{q_e}$ (k here: Boltzmann constant; T : temperature; q_e : electron charge) is a temperature dependent physical parameter independent of process technology or design choices. It is approximately 26mV at room temperature and we'll treat it as constant in the rest of this text, as we'll not dive into temperature dependencies of CMOS devices. However, be aware that CMOS devices are *very much* dependent on temperature and that dependency needs to be considered when designing products that will be exposed to different temperatures (as most products will be!).

Note that in the form equation (2.4) is given, all voltages appear as differences between two voltages, so it does not matter what voltage is defined as a zero/Gnd reference. However, sometimes the equation is given in a form where the bulk voltage is used as reference, i.e. $V_B := 0$. Then the above equation can be written as:

$$I_{F(R)} = I_S \ln^2 \left[1 + e^{\frac{V_G - nV_{S(D)} - V_t}{2nU_T}} \right] (1 + \lambda |V_D - V_S|) \quad (2.7)$$

But for now we'll stick with the form (2.4) where we keep V_B in the equation and the zero reference may be $-V_{ss}$, for example.

Equation (2.4) is the equation for *nFETs*! For pFETs one needs to inverse the differences in the equation, i.e. $V_{S(D)} - V_G$ instead of $V_G - V_{S(D)}$ and $V_{S(D)} - V_B$ instead of $V_B - V_{S(D)}$, and the sign of the expressions for I_F and I_R is inverted:

$$I_{F(R)} = -I_S \ln^2 \left[1 + e^{\frac{[V_{S(D)} - V_G] + (n-1)[V_{S(D)} - V_B] - V_t}{2nU_T}} \right] (1 + \lambda |V_D - V_S|) \quad (2.8)$$

Note that the naming of S and D is arbitrary for this symmetric device. However, by convention we shall assume in the following that for nFETs the S -terminal is the one with the lower voltage than the D terminal, and vice versa for pFETs. This way we ensure that $|I_F| > |I_R|$, and consequently the drain current flows *into* a nFET's drain terminal, whereas it flows *out* from pFET's drain terminal (i.e. the result from equation (2.8) will be negative).

Oftentimes one abbreviates $V_G - V_S - V_t$ as the **overdrive voltage** V_{OV} (e.g. in [2], while others call it the **effective voltage** V_{eff} [1], but we'll stick with V_{OV} in this compendium):

$$V_{OV} := V_G - V_{S(D)} - V_t \quad (2.9)$$

So one can rewrite (2.4) a bit more compactly:

$$I_{F(R)} = I_S \ln^2 \left[1 + e^{\frac{V_{OV} + (n-1)[V_B - V_{S(D)}]}{2nU_T}} \right] (1 + \lambda |V_D - V_S|) \quad (2.10)$$

Note that parameter λ is sometimes alternatively expressed as the Early Voltage $V_A = \frac{1}{\lambda}$ and V_A is proportional to the transistor length L and thus sometimes expressed as $V_A = V'_A L$, where $V'_A = \frac{1}{\lambda L} = \frac{1}{\lambda'}$ is a process parameter. There is a graphical way of determining V_A from a I_D vs V_{DS} plot (see figure 2.1): if one extends the slope of the curve in the saturation region all the way to where it intersects the V_{DS} -axis, the value of V_{DS} at that intersection is equal to $-V_A$.

Oftentimes one thinks of this formula as mostly determined by the difference of the gate voltage to the source voltage $V_{GS} := V_G - V_S$ and to a lesser degree by the difference of the drain to the source voltage $V_{DS} := |V_D - V_S|$, i.e. ignoring the dependency on V_{SB} which is completely appropriate if the so called slope factor $n = 1$.

$$I_{F(R)} = I_S \ln \left[1 + e^{\frac{V_{GS(GD)} - V_t}{2U_T}} \right]^2 (1 + \lambda(V_{DS})) \quad (2.11)$$

However, if $1 < n \leq 2$ and $V_S > V_B$, the same V_{GS} will now result in a smaller I_F . Then one can think of the difference of the bulk voltage to the source voltage having the same effect as the gate voltage to the source voltage, but with reduced efficiency by a factor $n - 1$. This is sometimes referred to as the **back gate** effect

This EKV model has been introduced to unify in a single equation the earlier models that used 4 different simpler formulas that are used dependent on regions of operation of the device. These regions depend on the voltages V_{GS} and V_{DS} as depicted in an example in figure 2.5. They are the four possible combinations of the two pairs of mutually exclusive regions denoted as **strong- vs weak inversion**, and **conduction vs saturation** (see later). When clearly within

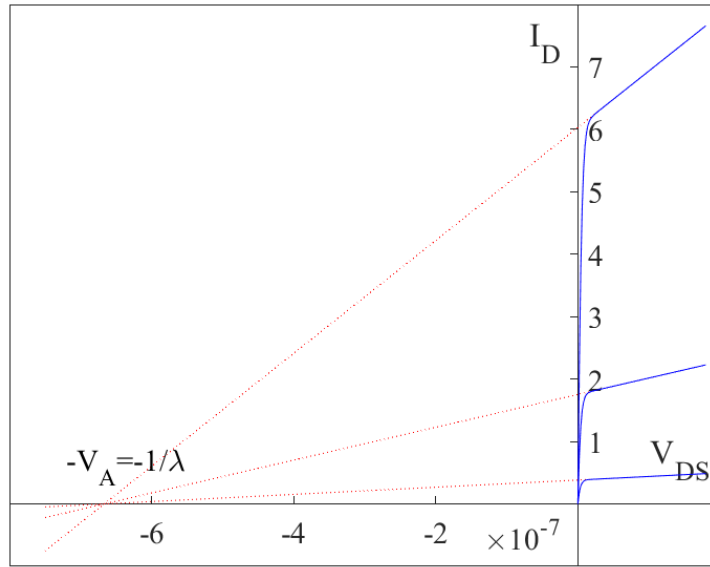


Figure 2.1: How to derive parameter λ graphically from a I_D vs V_{DS} plot: Extrapolating the straight line of the plot in the saturation region will cut the V_{DS} axis at $-V_A = -\frac{1}{\lambda}$.

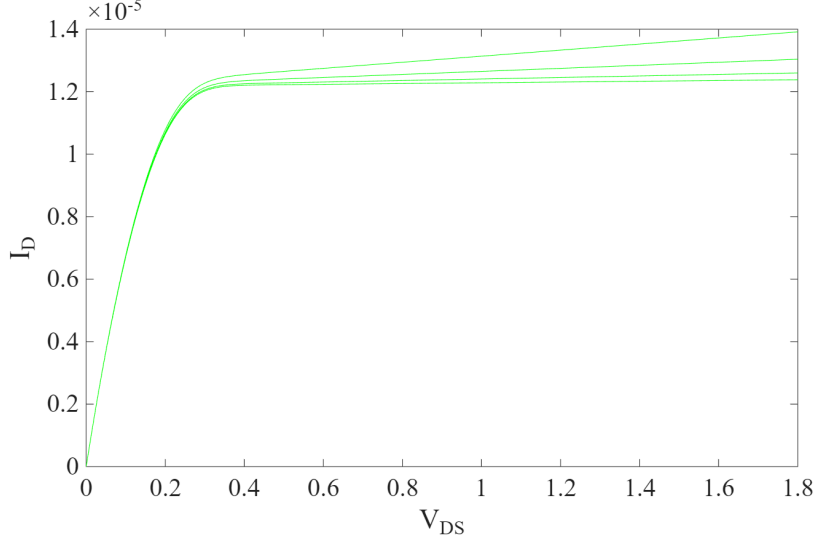


Figure 2.2: Typical I_D vs V_{DS} plots for varying λ . λ is the slope of I_D in the saturation region relative to the magnitude of I_D when $\lambda = 0$ (i.e. where the extension of the saturation region slope would hit the I_D -axis). The parameters used have been gleaned from a 180nm process and were $k = 270 \frac{\mu A}{V^2}$, $\frac{W}{L} = 1$, $V_t = 0.45V$, $U_T = 26mV$, $n = 1$, $V_{GS} = 0.75V$ (i.e. $V_{OV} = 0.3V$). λ was varied $\lambda = \{0.01, 0.02, 0.04, 0.08\}$

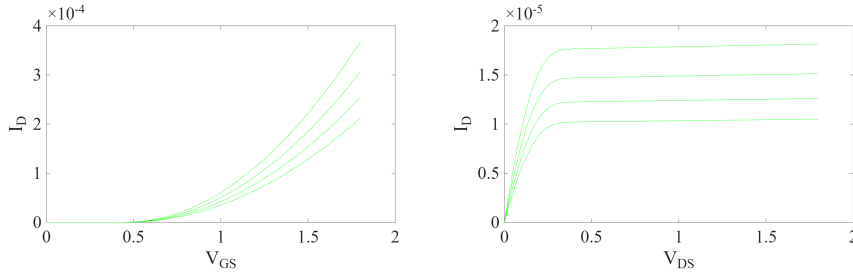


Figure 2.3: Typical I_D vs V_{GS} and V_{DS} plots for varying $I_S = 2nkU_T^2$. The parameters used have been gleaned from a 180nm process and were $V_t = 0.45V$, $U_T = 26mV$, $n = 1$. For the V_{GS} -plot $V_{DS} = 1.8V$. For the V_{DS} -plot, $V_{GS} = 0.75V$ (i.e. $V_{OV} = 0.3V$). I_S was varied as $I_S = \{304.2, 365.0, 438.0, 525.7\}nA$. Note that in a linear V_{GS} -plot, one cannot really see the weak inversion currents, as they are smaller than I_S , i.e. orders of magnitude smaller than the highest strong inversion currents. A Logarithmic I_D -axis would reveal those currents (see figure 2.6) for an example.

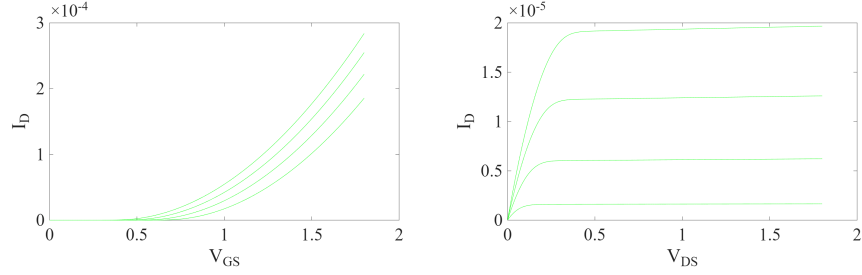


Figure 2.4: Typical I_D vs V_{GS} and V_{DS} plots for varying V_t . The parameters used have been gleaned from a 180nm process and were $k = 270 \frac{\mu A}{V^2}$, $\frac{W}{L} = 1$, $U_T = 26mV$, $n = 1$, $V_{GS} = 0.75V$ (i.e. $V_{OV} = 0.3V$). For the V_{GS} -plot $V_{DS} = 1.8V$. For the V_{DS} -plot, $V_{GS} = 0.75V$ (i.e. $V_{OV} = 0.3V$). V_t has been varied as $V_t = \{0.375, 0.450, 0.540, 0.648\}V$.

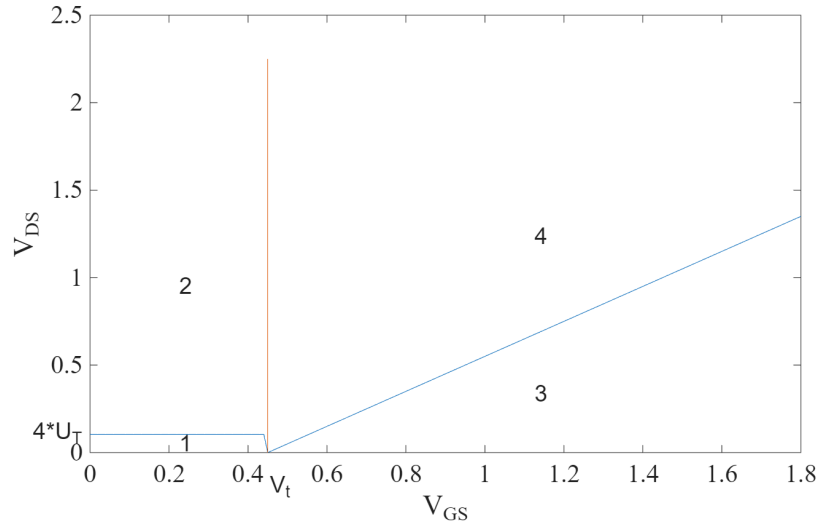


Figure 2.5: Four regions of operation where the simplified FET large models can be applied. However, the full model has to be used when near the transitions of these regions. 1) weak inversion and conduction; 2) weak inversion and saturation; 3) strong inversion and conduction; 4) strong inversion and saturation.

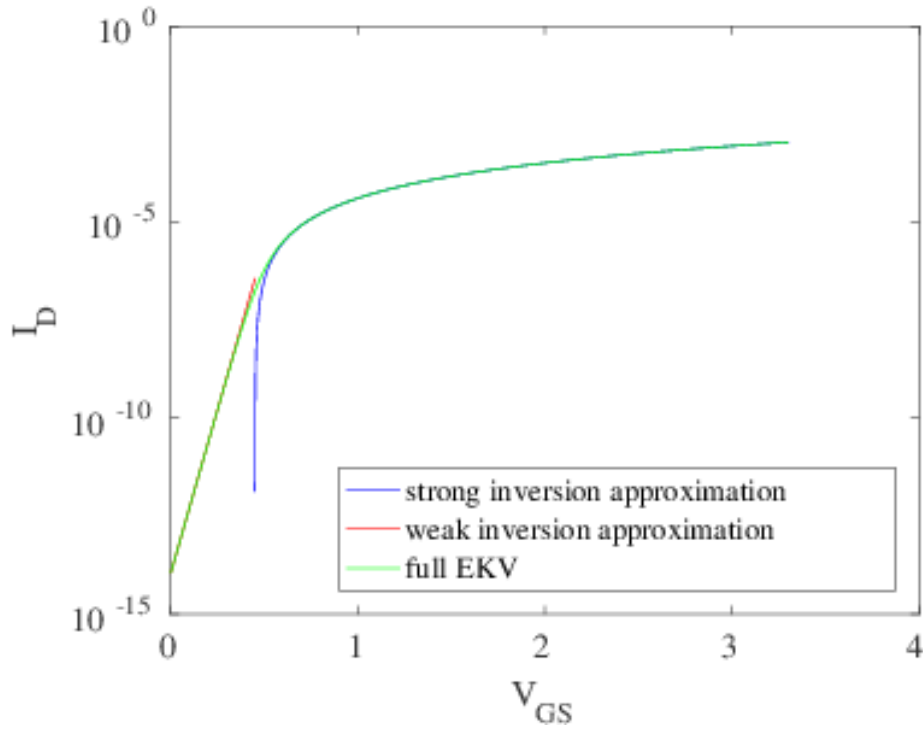


Figure 2.6: Illustrating the shortcomings of the simplified models in specific regions of operation. Here we show an I_D vs V_{GS} plot with a logarithmic I_D -axis, to show both the weak and strong inversion region clearly. The simplified models in red and blue, are discontinuous and non-differentiable at the point of separation V_t between weak- and strong inversion here. Only the full EKV in green offers a smooth transition and correct results in *moderate* inversion.

one of those regions of operation, the EKV model converges to those simpler formulas. They are easier to use for hand calculations. However, be careful when close to the borders of these regions. Then the simplified equations will be quite wrong and the full formula (2.4) has to be used. Figure 2.6 illustrates the danger of relying on these simplified models when being in **moderate inversion** (i.e. close to the separation of strong- vs weak inversion), where especially the strong inversion approximation, can give very wrong results. Similarly when operating with V_{DS} close to the saturation voltage V_{sat} , the results will be inaccurate. So always verify that the simplified models can actually be applied, i.e. that you are a good way away from the separating voltages, i.e. from V_t for V_{GS} and from V_{sat} for V_{DS} when using those models!

As introduced before V_t is a constant and it is the point of separation for the gate to source voltage V_{GS} separating the **weak inversion** region from the **strong inversion** region:

$$V_{GS} \begin{cases} \ll V_t : \text{weak inversion rules can be used} \\ \approx V_t : \text{moderate inversion: full EKV needs to be used} \\ \gg V_t : \text{strong inversion rules can be used} \end{cases} \quad (2.12)$$

Equivalent condition for strong inversion is that either I_F or I_R is bigger than $0.48 * I_S$ (or often just simply I_S is used)

$$I_F \begin{cases} \ll 0.48 I_S : \text{weak inversion rules can be used} \\ \approx 0.48 I_S : \text{moderate inversion: full EKV needs to be used} \\ \gg 0.48 I_S : \text{strong inversion rules can be used} \end{cases} \quad (2.13)$$

The saturation voltage V_{sat} is not always constant. It depends on whether the device is operated in strong or weak inversion:

$$V_{sat} = \begin{cases} 4U_T & \text{if } V_{GS} \ll V_t \\ 4U_T & \text{if } V_{GS} \approx V_t \text{ (i.e. use } 4U_T \text{ as a minimum until } \frac{V_{GS}-V_t}{n} > 4U_T) \\ \frac{V_{GS}-V_t}{n} & \text{if } V_{GS} \gg V_t \end{cases} \quad (2.14)$$

Its the point of separation on the V_{DS} axis between the **conduction** region and the **saturation** region:

$$V_{DS} \begin{cases} \ll V_{sat} : \text{conduction rules can be used} \\ \approx V_{sat} : \text{full EKV needs to be used} \\ \gg V_{sat} : \text{saturation rules can be used} \end{cases} \quad (2.15)$$

So these two separators leave us with 4 distinct regions as depicted in figure 2.5. Here are the simplified equations one can employ if clearly within one of these regions (but not (!) if one is close to the transition between those regions!). Note that these following equations are written in the form where all voltages are defined relative to $V_B := 0$ as zero reference potential.

strong inversion, saturation, nFET

Conditions for it to be applicable:

1. $V_{GS} \gg V_t$ or equivalent $I_F \gg 0.48I_S$ (condition for strong inversion)
2. $V_{DS} \gg V_{sat} := \frac{V_{GS}-V_t}{n}$ ($\underline{\Delta}$: different for weak inversion) or equivalently $I_F \gg I_R$ (condition for saturation)

Simplified equation for this region:

$$I_D = \frac{1}{2n} k_n (V_G - V_{tn} - nV_S)^2 (1 + \lambda V_{DS}) \quad (2.16)$$

I_D vs V_{GS} is a square function where:

$$I_D \propto (V_G - nV_S - V_t)^2 \quad (2.17)$$

I_D vs V_{DS} is a linear function where:

$$I_D \propto (1 + \lambda V_{DS}) \quad (2.18)$$

A square law governs the I_D vs V_{GS} behaviour. Many older books do in fact only discuss the strong inversion cases as weak inversion currents were considered negligibly small.

The dependency of I_D vs V_{DS} is linear: $I_D \propto (1 + \lambda V_{DS})$ (both strong and weak inversion). For $\lambda = 0$ the V_{DS} curve is completely flat, i.e. the FET is behaving like an ideal current source producing the same current irrespective of the voltage V_{DS} .

weak inversion, saturation, nFET

Conditions for it to be applicable:

1. $V_{GS} \ll V_t$ or equivalent $I_F \ll 0.48I_S$ (condition for weak inversion)
2. $V_{DS} \gg V_{sat} := 4U_T$ ($\underline{\Delta}$: different for strong inversion) or equivalently $I_F \gg I_R$ (condition for saturation)

Simplified equation for this region:

$$I_D = I_S e^{\frac{V_G - V_{tn} - nV_S}{nU_T}} (1 + \lambda V_{DS}) \quad (2.19)$$

I_D vs V_{GS} is an exponential function where :

$$I_D \propto e^{\frac{V_G - nV_S - V_t}{nU_T}} \quad (2.20)$$

I_D vs V_{DS} is a linear function where:

$$I_D \propto (1 + \lambda V_{DS}) \quad (2.21)$$

So in weak inversion in saturation for I_D vs V_{GS} this exponential law applies ($I_D \propto e^{\frac{V_{GS}-V_t}{nU_T}}$).

And for I_D vs V_{DS} we get the same proportional linear relationship ($I_D \propto (1 + \lambda V_{DS})$) like in strong inversion, i.e. a current source-like behaviour for small λ .

strong inversion, conduction

Conditions for it to be applicable:

1. $V_{GS} \gg V_t$ or equivalent $I_F \gg 0.48I_S$ (condition for strong inversion)
2. $V_{DS} \ll V_{sat} := \frac{V_{GS}-V_t}{n}$ ($\underline{\Delta}$: different for weak inversion) or equivalently $I_F \approx I_R$ (condition for conduction)

Simplified equation for this region:

$$I_D = k_n V_{DS} \left[V_G - V_{tn} - \frac{n}{2}(V_D + V_S) \right] \quad (2.22)$$

I_D vs V_{GS} is a linear function.

I_D vs V_{DS} in deep conduction is a linear function where for $V_S = 0$ and $V_{DS} \ll V_{OV}$ the first order Taylor expansion in V_{DS} results in the behaviour of a resistor:

$$I_D = k_n V_{ov} V_{DS} \Rightarrow \frac{1}{r_{ds}} = g_{DS} = k_n V_{ov} \quad (2.23)$$

So in the conduction region (both strong and weak inversion) the I_D vs V_{DS} behaves very much like a resistor, and one that can be tuned by V_{GS} at that.

weak inversion, conduction

Conditions for it to be applicable:

1. $V_{GS} \ll V_t$ or equivalent $I_F \ll I_S$ and $I_R \ll I_S$ (condition for weak inversion)
2. $V_{DS} \ll V_{sat} := 4U_T$ ($\underline{\Delta}$: different for strong inversion) or equivalently $I_F \approx I_R$ (condition for conduction)

Simplified equation for this region:

$$I_D = I_S e^{\frac{V_G - V_{tn}}{nU_T}} \left(e^{\frac{-V_S}{U_T}} - e^{\frac{-V_D}{U_T}} \right) \quad (2.24)$$

I_D vs V_{GS} is an exponential function.

I_D vs V_{DS} in deep conduction is a linear function, where for I_D vs V_{DS} and $V_S = 0$ close to the origin $V_{DS} \ll V_{sat}$ (1st order Taylor expansion around $V_{DS} = 0$):

$$I_D = I_S e^{\frac{V_{ov}}{nU_T}} \frac{V_D}{U_T} \Rightarrow \frac{1}{r_{ds}} = g_{DS} = I_S e^{\frac{V_{ov}}{nU_T}} \frac{1}{U_T} \quad (2.25)$$

So a resistive behaviour $r_{DS} = \frac{1}{g_{DS}}$

Note that there are other names in use for those regions: Strong inversion is also called '**above threshold**' and weak inversion '**subthreshold**'. The saturation region is also referred to as '**active**' region, and the conduction region as '**triode**' or '**linear**' region.

2.2 Capacitances

Looking at figure 1.1 and 1.2 in search of capacitances that add to the model one can find two types of structures that cause capacitors: Conductors separated by an insulator, and reversed biased PN-junctions (i.e. conductive regions separated by a depletion region). In figure 1.1 the depletion regions for the transistors in saturation are sketched as grey areas. So they give rise to a capacitance between source and bulk C_{SB} , and drain and bulk C_{DB} . The SiO_2 insulating the gate from the channel is responsible for C_{GS} , as well as (generally smaller) C_{GD} . One gets the same capacitances also for a silicon on insulator FET (figure 1.2). However, it is the buried silicon dioxide (BOX) instead of a depletion region that governs C_{SB} and C_{DB} , which generally are smaller.

The plate capacitors' capacitance depends on a technology dependent capacitance per area, i.e. C_{OX} for the gate oxide and C_{BOX} for the buried oxide. Most straight forward is C_{GD} which is:

$$C_{GD} = L_{OV}WC_{OX} \quad (2.26)$$

where L_{OV} is the length of the overlap between the gate and the drain, also taking into account some 'length' accounting for the fringe capacitance along the edge of the gate towards the drain. This length is usually a constant value for a given technology, independent of the transistor dimensions. So note the take home message: C_{GD} increases with W .

A bit more tricky is C_{GS} . It is NOT actually constant and depends on all the transistor terminal voltages. This compendium, however, is not covering this and uses a common approximate voltage independent value:

$$C_{GS} \approx \left(\frac{2}{3}WL + WL_{OV} \right) C_{OX} \quad (2.27)$$

For SiO technologies (Fig. 1.2) the capacitances towards the bulk are functions of C_{BOX} and the source and drain areas respectively. For bulk technologies its the reversed biased PN-junctions as well as the depletion region beneath the channel. Junction capacitance in reversed biased diodes is mostly dependent on the doping profile, but also not a constant for different voltages applied. This compendium is not dealing with these voltage dependencies either, but to give the students at least a qualitative understanding of the dependency on doping: higher doping differences lead to more narrow depletion regions and thus higher capacitance. Generally when doing circuit design this is not something the designer can choose, so we can only modify the source and drain areas to influence these capacitances. If we assume a fixed junction capacitance per area C_J at source and drain and a channel to bulk capacitance per area C_{chB} , we get:

$$C_{DB} = C_JA_D \quad (2.28)$$

$$C_{SB} \approx C_JA_S + \frac{2}{3}WLC_{chB} \quad (2.29)$$

If the layout of the S/B terminals is rectangular with a width equal to the channel width W and minimal constant length $L_{S/D}$ then these capacitances are proportional with W as:

$$A_{S/D} = WL_{S/D} \quad (2.30)$$

Note that often times it is important to also keep an eye out for other parasitic capacitances, e.g of interconnections between transistors, for overall circuit performance. Also metal wires, if not kept very short, will add capacitance to an electric node (we will use C_W later in this compendium to refer to wiring capacitance). If they are very long they need to be treated using cable equations, i.e. they are not a single electrical node but signals travelling along a cable can be changed along the way and reflected at the cable termination. This is again not something this compendium will cover, but can none the less be important when designing high performance/frequency (digital or analog) circuits.

Chapter 3

FET Small Signal Models

Small signal models are used in analogue circuit analysis. They are linearised models of the approximate behaviour of a circuit around a particular **point of operation (PO, aka biasing point)** mainly used for analysis of analogue circuits. Linear circuit theory has over the decades accumulated a lot of methods and tools for circuit analysis. Unfortunately the model for FETs we have introduced so far is anything but linear. However, in the following we will introduce a linear approximation for the FET behaviour using 1st order Taylor expansion around a **point of operation**. This linearised model can then only model the behaviour with any accuracy for **small signal** variations around this point of operation, in contrast to the full scale or **large signal** model we have introduced before.

3.1 FET behaviour linear approximation around a point of operation

A general expression for a first order Taylor expansion $i_D(v_G, v_S, v_D)$ of I_D in variables V_{GS} and V_{DS} of function (2.2) around a point of operation is:

$$i_{DS} = I_{DS}(V_{GS}, V_{DS}) + \delta v_{GS} * \frac{\delta I_{DS}}{\delta V_{GS}} + \delta v_{DS} * \frac{\delta I_{DS}}{\delta V_{DS}} \quad (3.1)$$

Look at figure 3.1 for an illustration. The blue lines are the large signal model and the red lines are the linear approximation around the DC point of operation/biasing point marked with the red circles expressed by equation 3.1.

Note here that we use upper case variable names and upper case indices for *biasing voltages/currents*, i.e. the values of the point of operation that are derived by large signal models and lower case variables with upper case indices for the linearized signals around that point of operation/biasing point. δ indicates an infinitesimally small deviation from a point of operation. As a further simplification of the notation we will also refer to the infinitesimally small variations indicated by the δ plus variable name with using lower case variables and index letters, so for example $v_{gs} := \delta v_{GS}$. These latter signals we refer to as the **small signal variables** or simply **small signals**. These small deviations from the point of operation are illustrated in the figure by the green

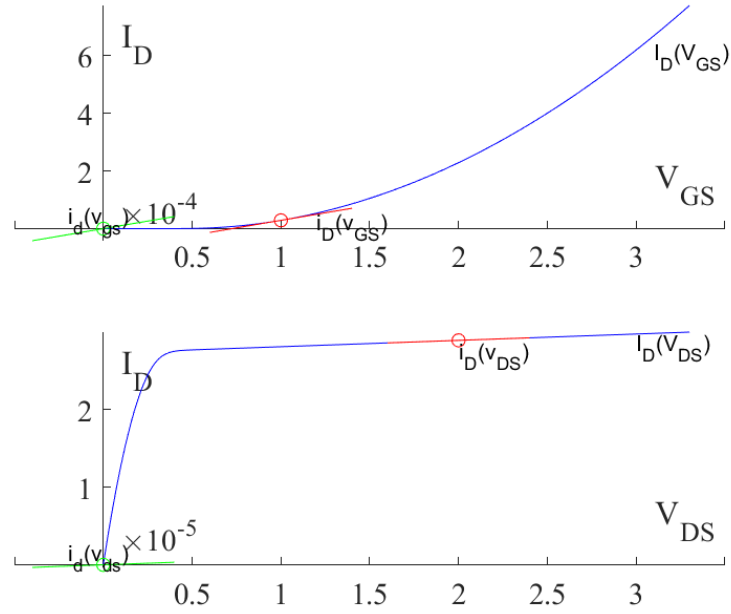


Figure 3.1: Illustrating the linear approximation of the nFET large signal model in a particular **point of operation/biasing point**. In blue there is the large signal behaviour, in red the linearized behaviour around an (arbitrarily chosen as $V_{GS} = 1V$ and $V_{DS} = 2V$) point of operation/biasing point, and in green the small signal behaviour (i.e. the same as the linearized behaviour but 'ignoring' the biasing point/DC offset/point of operation). This is for an assumed 180nm CMOS technology with $\mu_n C_{ox} = 270 \frac{\mu A}{V}$, $V_{dd} = 3.3V$, $V_t = 0.45V$, $n = 1.5$, $\lambda = 0.03 \frac{1}{V}$

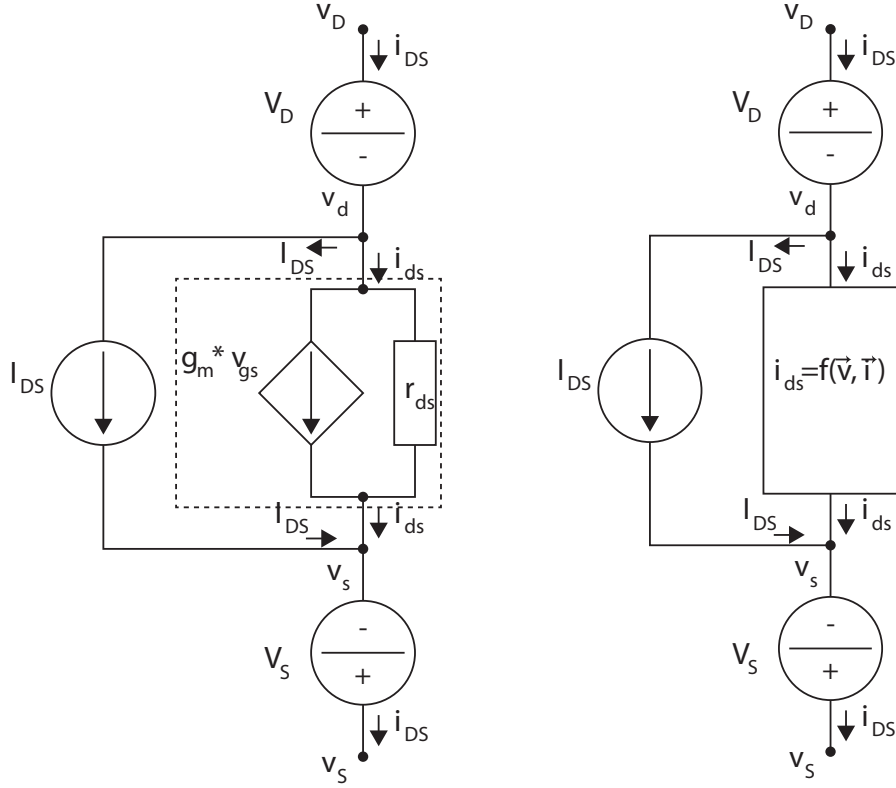


Figure 3.2: Left: Linear circuit approximation of a FET in a particular point of operation/biasing point. Right: General two terminal linear circuit model with explicit DC-offset, i.e. point of operation. In general, f would be a linear operator of all network voltage signals and current signals, but for now we shall think of it as a function.

lines, i.e. *only* the small variations, without the point of operation. So rewriting the same equation with these notation becomes:

$$i_{DS} = I_{DS} + v_{gs} * \frac{\delta I_{DS}}{\delta V_{GS}} + v_{ds} * \frac{\delta I_{DS}}{\delta V_{DS}} \quad (3.2)$$

Furthermore the two derivatives in the equation are now parameters of this new linearised function derived from the large scale model. For further simplification of the notation we do give those two parameters their own symbols, i.e. $g_m := \frac{\delta I_{DS}}{\delta V_{GS}}$ and $g_{ds} := \frac{1}{r_{ds}} := \frac{\delta I_{DS}}{\delta V_{DS}}$. So rewriting the same function once again:

$$i_{DS} = I_{DS} + v_{gs} * g_m + v_{ds} * \frac{1}{r_{ds}} \quad (3.3)$$

We can then draw a linear circuit that models the linear model current i_{DS} that can be used in a linear circuit model in lieu of a transistor between the terminals (also resulting from interaction in a linear version of the overall circuit this FET might be part of) v_D and v_S as drawn in the left of figure 3.2. Note that the circuit in the dashed box on it's own behaves just like the entire circuit

does, but with only the small signals at its terminals. In other words it behaves like the complete circuit but its point of operation is defined by all bias voltages and currents being zero. The circuit in the dashed box is called the **small signal equivalent circuit element**.

3.2 Small signal equivalent circuit

The left of figure 3.3 shows an example of a small network of such circuit elements linearized around a biasing point. Note that all the bias currents are derived from solving the Kirchhoff and Ohm equations for the large signal models at that point, so $I_3 = I_1 + I_2$ and $I_3 = I_4$. Also, the small signal deviations from the biasing point need to fulfil the same requirement for the small signals $i_3 = i_1 + i_2$ and $i_3 = i_4$. In a network of general circuit elements that are linearised this way for a specific point of operation (see example in the left of figure 3.3) one can 'transpose' the point of operation defined by the two input voltages and the biasing current to all of those three parameters being zero: The bias voltage sources as well as the bias current sources cancel each other out in points v_3 and v_4 . So one can remove the voltage sources and disconnect the current sources and the terminal currents and voltages for the small signal circuit element i_3 are still the same. One is left with the network on the right: it behaves exactly the same in all other nodes, but now for nodes v_3 and v_4 and the current between them i_3 only the small signals, i.e. the deviations from the biasing point, are modelled.

This brings us to a last simplification used in the so called small signal analysis of a circuit: when analysing the small signal behaviour of a circuit around a particular point of operation one does replace **ALL** the circuit elements with these small signal equivalent circuits (i.e. the 1st order Taylor equivalent with all DC sources removed), i.e. one does set all bias/DC sources to zero (disconnecting DC currents and shorting DC voltages). The analysis of the resulting small signal equivalent circuit does only model the linearised *deviations* from the point of operation, i.e. if one wants the entirety of the signal back, one has to add the DC point of operation values to the small signal values again.

Note that the basic circuit elements, namely resistors, capacitors, and inductors, already *ARE* linear circuit elements (see definition of a linear system in section 5.1) and their small signal equivalent circuits simply are themselves. That is because for example the total current through a resistor can be simply linearly separated into a large signal DC component and a small signal offset.

$$I_R + i_r = \frac{V_R}{R} + \frac{v_r}{R} \quad (3.4)$$

$$I_C + i_c = \frac{\partial}{\partial t} V_C C + \frac{\partial}{\partial t} v_c C \quad (3.5)$$

$$V_L + v_l = \frac{\partial}{\partial t} I_L L + \frac{\partial}{\partial t} i_l L \quad (3.6)$$

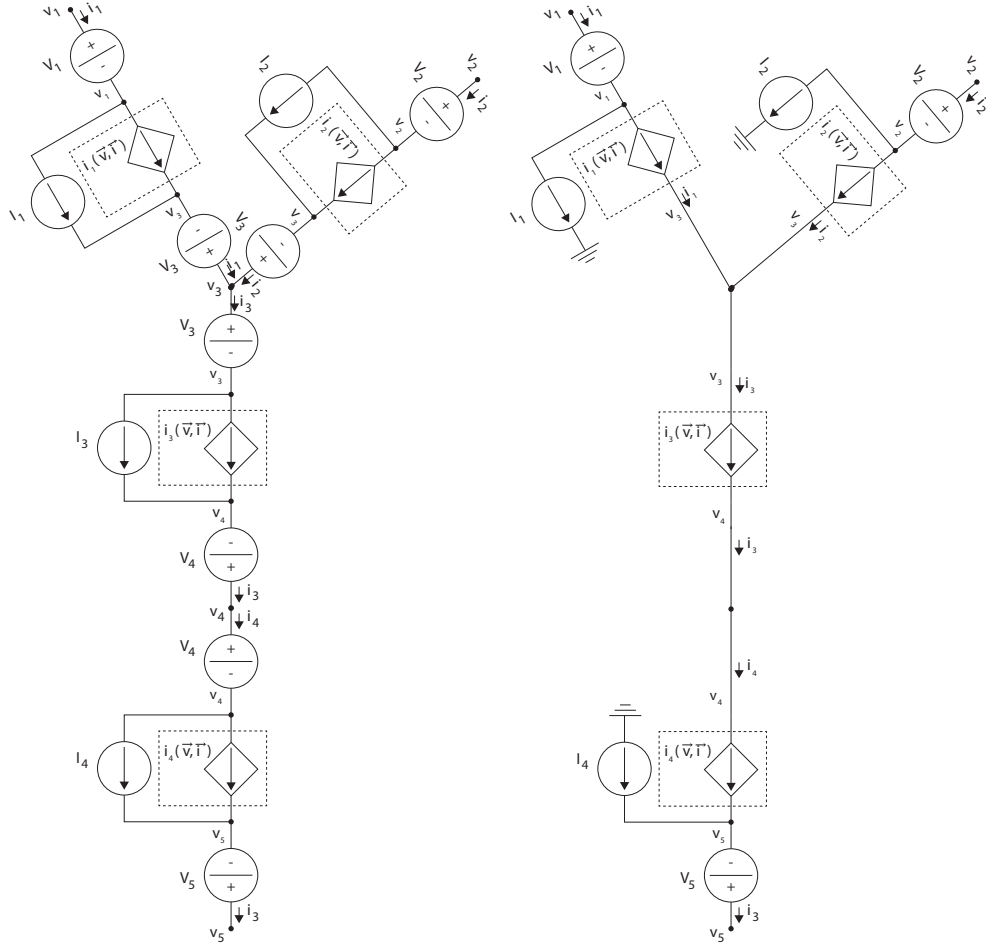


Figure 3.3: Left: A electronic circuit network of linear approximation elements Right: Replacing the element in the center with its small signal equivalent that neglects any DC offsets and does not affect the behaviour of the rest of the network. This is to illustrate that one can in fact replace any and all elements with the small signal equivalents, i.e. ignore any DC offsets/biasing points and still model the linear changes in the network correctly.

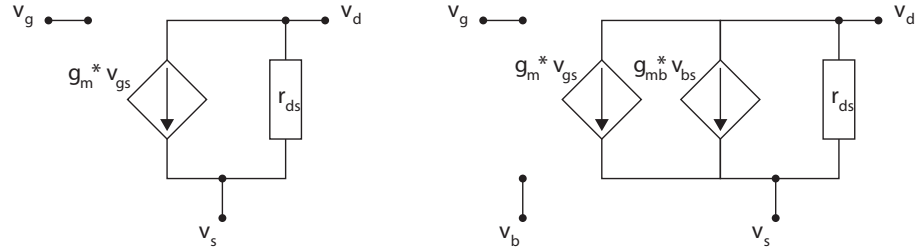


Figure 3.4: FET Small signal equivalent model for low frequencies. g_m is the **transconductance** and r_{ds} is the **drain resistance**. The right hand side version includes the so called **back gate effect**, i.e. the influence of varying the bulk to source voltage. The **back gate transconductance** g_{mb} is usually about 10 times smaller than g_m and often neglected.

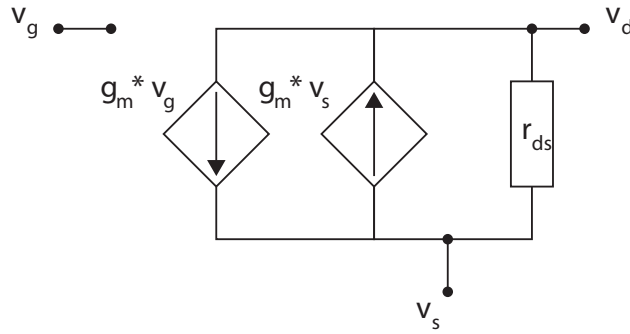


Figure 3.5: The same as the left hand side in figure 3.4, but explicitly separating the influence of respectively v_g and v_s . This is sometimes a bit less prone to making sign errors when deriving small signal equivalent circuits where one of the two is constant and can thus be removed.

3.3 FET low frequency small signal model

In a first instance, let us look at the **low frequency small signal equivalent** for FETs. Low frequency means that we'll for now neglect 'parasitic' capacitors in these models that would introduce dependencies on the *rate of change* of dynamic signals. The appropriate model already mentioned in figure 3.2 is given without distraction in figures 3.4 and 3.5.

The model parameters g_m (the **transconductance**) and r_{ds} (the **drain resistance**) need to be derived for a specific **point of operation (PO)** from the derivatives of the large signal model. The derivation of g_m in dependency of I_D does not depend on whether the device is in saturation or conduction. Only if it is in strong or weak inversion. (However, note that if you derive the dependency on V_{OV} instead of I_D THAT will depend on whether the device is in saturation or conduction!)

g_m for strong inversion, saturation:

$$g_m = \frac{dI_D}{dV_{GS}} = k_n V_{OV}(1 + V_{DS}\lambda) = \sqrt{2k_n I'_D}(1 + V_{DS}\lambda) = \sqrt{2k_n I_D}(1 + V_{DS}\lambda) \quad (3.7)$$

The term $(1 + V_{DS}\lambda) \approx 1$ is often neglected, so $I_D \approx I'_D$. I'_D is the **saturation current** neglecting channel modulation, i.e. setting $\lambda = 0$ or you can define:

$$I'_D := \frac{I_D}{1 + V_{DS}\lambda} \quad (3.8)$$

g_m for strong inversion, conduction:

$$g_m = \frac{dI_D}{dV_{GS}} = k_n V_{DS} \quad (3.9)$$

g_m for weak inversion:

$$g_m = \frac{dI_D}{dV_{GS}} = \frac{I_D}{U_T} \quad (3.10)$$

r_{ds} in dependency of I_D in saturation does not depend on whether the FET is biased in weak or strong inversion and is:

r_{ds} in saturation:

$$r_{ds} = \frac{1}{\lambda I'_D} = \frac{V_A}{I'_D} \quad (3.11)$$

In deep conduction ($V_{DS} \ll V_{OV}$), it *does* depend on whether one is in weak or strong inversion, though:

r_{ds} in deep conduction, strong inversion:

$$r_{ds} = \frac{1}{k V_{OV}} \quad (3.12)$$

r_{ds} in deep conduction, weak inversion:

$$r_{ds} = \frac{1}{I_S U_T} e^{-\frac{V_{OV}}{n U_T}} \quad (3.13)$$

Note: since these equations only can be applied when clearly within a region of operation, you need to be careful when close to the point of transition

between these regions. You then better apply a numerical approximation of the derivatives using the full EKV model, or rely on simulation tools entirely.

The bulk B exerts a similar effect on the channel as the gate and one can thus also derive a transconductance g_{mb} for the linearized effect of V_B on I_D : it is also coupled capacitively to the channel, after all. Note the restriction that in bulk CMOS V_B cannot move above V_S for nFETs and not below V_S for pFETs, to avoid a forward biased PN junction between them(!). So this effect can be used more freely in SOI technologies, where this restriction does not occur. The ability of V_B to modulate I_D is sometimes referred to as **back gate effect**. As the depletion region and BOX are not as thin as the gate oxide, the effect is minor by comparison with the effect of V_G . Specifically, V_B is $(n - 1)$ as effective in controlling the drain current according to (2.4), or as a rule of thumb $\frac{1}{10}$ as effective if n is not known. Oftentimes the bulk is connected to the source though, and g_{mb} does thus not have any effect. If the source is at a higher potential than the bulk however, one has to be aware that the transistor will conduct less current for the same V_{GS} in the large signal world and that v_b can be used to modify the drain current in the small signal world.

$$g_{mb} = (n - 1)g_m \approx g_m \frac{1}{10} \quad (3.14)$$

3.4 Applying small signal analysis

So to summarize the recipe for replacing a circuit by it's small signal equivalent to determine its behaviour around a particular biasing point:

1. Fully determine the point of operation, i.e. all biasing parameters.
2. Derive small signal parameters (e.g. g_m and r_{ds}) based on that point of operation.
3. Replace all circuit elements with their small signal equivalent.
4. Remove all independent/constant current sources and replace all independent/constant voltage sources with Gnd (i.e. NOT the current sources that depend on a small signal input such as the $i_{ds} = g_m v_{gs}$).
5. perform a normal Kirchhoff's or Ohm's law analysis of the circuit.

Example

A simple first example of such a small signal analysis of the so called common source (CS) amplifier is shown in figure 3.6. An example of it's voltage I/O relationship is shown in figure 3.7. In the middle of its input range the output voltage falls steeply. If the circuit is used as an amplifier, this part is the interesting **input voltage range**. The best point of operation/biasing point is the red circle smack in the middle of the steep signal transition. If one follows the red linearised model, the output voltage v_O is given as $v_O = Av_I$ where $A < -1$ is a negative voltage amplification.

So for the small signal analysis we now have to first find all the correct biases to have the circuit operate in this point of operation. For that, we have to use

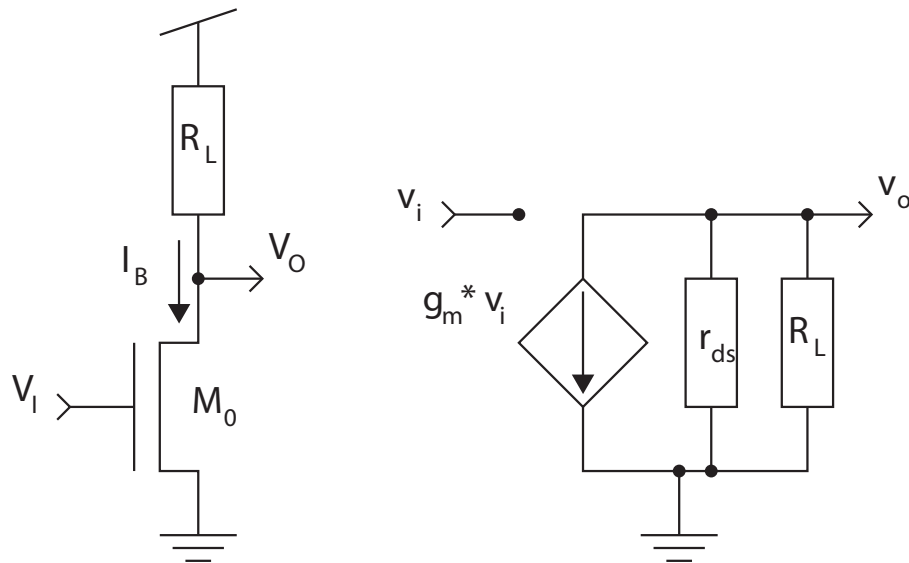


Figure 3.6: CS amplifier circuit and its small signal equivalent circuit.

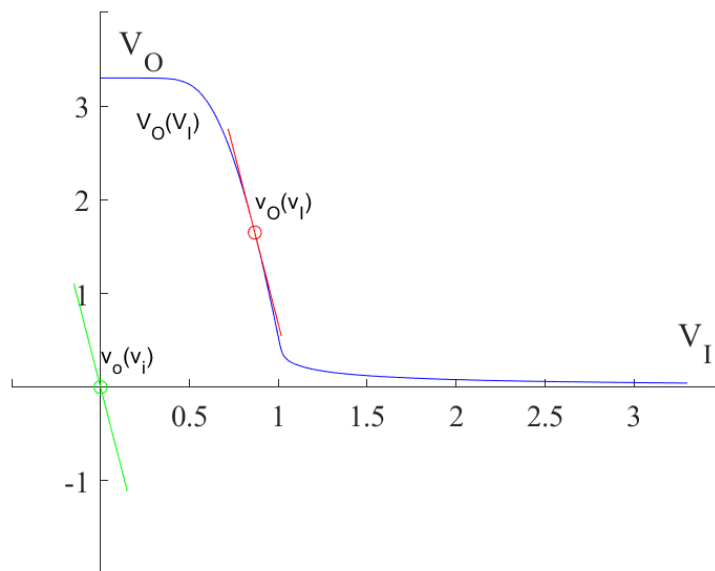


Figure 3.7: CS stage I/O relationship according to the large signal model (blue), the linearised model (red), and the small signal model (green).

the large signal model! Once we do have found all these biasing parameters to end up in the correct point of operation we can then find the correct small signal parameters for that point of operation and finally find the correct number for A .

Step 1: find biasing point, i.e. all biasing parameters.

To find/set the point of operation we move in the large signal model world. Some assumptions/parameters need to be defined/given in the task to find those biasing voltages and the current. As process parameters and design parameters the transistor has a characteristic current $I_S = 50\text{nA}$, a threshold voltage $V_{tn} = 0.5\text{V}$, a $\lambda = 0.05\frac{1}{\text{V}}$, and a negligible slope factor $n = 1$. At room temperature we use $U_T = 26\text{mV}$. The bulk is shorted to the source $V_S = V_B = 0\text{V}$. The supply voltage is 3.3V and the load resistance is given $R_L = 1\text{M}\Omega$. Furthermore the task is given the point of operation for the output voltage V_O to be at $\frac{V_{dd}}{2}$.

In summary:

Given: the process and design parameters (I_S , V_{tn} , λ , n , U_T , V_{dd} , V_B), V_O and R_L

Asked: I_B , V_I , and A

So the voltage fall across the resistor has to be 1.65V . From that it follows directly that the bias current I_B needs to be $I_B = 1.65\mu\text{A}$. This will put the output voltage V_O neatly between Gnd and V_{dd} at 1.65V . The last parameter that is not yet defined is the point of operation for the input voltage V_I . It is fully dependent on I_B and V_O and can be derived from them. First let's make the observation that $I_B \gg I_S$. That is one way of determining that the strong inversion model simplification is appropriate in this case. We also suspect and assume (to be verified) that the transistor is in saturation. So we shall use equations (2.16) and (2.5) to derive V_I .

$$V_G = 2nU_T \sqrt{\frac{I_D}{I_S (1 + \lambda V_{DS})}} + V_{tn} + nV_S \quad (3.15)$$

Entering the already known parameters we get:

$$V_I = 2 * 26\text{mV} \sqrt{\frac{1.65\mu\text{A}}{50\text{nA} (1 + 0.05\frac{1}{\text{V}} * 1.65\text{V})}} + 0.5\text{V} \quad (3.16)$$

One should always be **very careful with the units!!!** Do not mix 'micro-', 'milli-', 'nano-', 'mega-' and the like. I highly recommend to always write the units into the equation in order to be always aware of what unit is being used. Also, at some point one has to make sure to use consistent units everywhere in the equation. For example by getting rid of all the prefixes, preferably right from the start:

$$V_I = 2 * 26 * 10^{-3}\text{V} \sqrt{\frac{1.65 * 10^{-6}\text{A}}{50 * 10^{-9}\text{A} (1 + 0.05\frac{1}{\text{V}} * 1.65\text{V})}} + 0.5\text{V} \quad (3.17)$$

One other advantage of writing the units into the equation is that one can verify that the end result actually gets the correct unit, [V] in this case. So that's a good control

$$V_I = 0.79\text{V} \quad (3.18)$$

Let us **not forget** to verify that we are in saturation here and have thus used the right equation. In strong inversion $V_{sat} = V_{OV}$. The overdrive voltage $V_{OV} = V_{GS} - V_{tn} = 0.29\text{V}$ and is much smaller than $V_{DS} = 1.65\text{V}$, so we *are* in saturation, the simplified equations are thus consistent with the emerging situation, and thus we are fine.

Step 2: find small signal parameters.

Let's move on to determine the small signal parameters g_m and r_{ds} at that point of operation. We use equations (3.7) and (3.11):

$$g_m = \sqrt{2k_n I_D} = \sqrt{2 \frac{I_S I_D}{2nU_T^2}} = \frac{\sqrt{50 * 10^{-9}\text{A} * 1.65 * 10^{-6}\text{A}}}{26 * 10^{-3}\text{V}} = 11.05 \frac{\mu\text{A}}{\text{V}} \quad (3.19)$$

$$r_{ds} = \frac{1}{\lambda I_D} = \frac{1}{0.05 \frac{1}{\text{V}} * 1.65 * 10^{-6}\text{A}} = 12.12\text{M}\Omega \quad (3.20)$$

Step 3 and 4: derive small signal equivalent circuit (replace elements and remove DC sources)

The small signal equivalent circuit is shown in figure 3.6 to the right. Note that there actually are *three* circuit elements to start with: the FET, the resistor *and* the voltage supply (i.e. an implicit DC voltage source for Vdd). Vdd is a DC voltage source that becomes Gnd in the small signal world. Thus r_{ds} and R_L actually end up in parallel between the transistor drain and Gnd. Also the DC offset at the input V_I and V_O at the output are no longer represented in this small signal model, nor is the bias current I_B that flows in the large signal representation only. What it models is the green line in figure 3.7, moving around the origin (the green circle) if any of the small signal variables is moving away from zero.

Step 5: Kirchhoff's and/or Ohm's rule to analyse

If you move v_i you'll create a current $g_m * v_i$ flowing from Gnd through the parallel resistors *towards* v_o , i.e. v_o has the inverted sign compared to v_i . More precisely:

$$v_o = -g_m * v_i (r_{ds} || R_L) \quad (3.21)$$

and

$$A = \frac{v_o}{v_i} = -g_m (r_{ds} || R_L) = -g_m \frac{r_{ds} R_L}{r_{ds} + R_L} \quad (3.22)$$

Since $r_{ds} \gg R_L$ it follows that $(r_{ds} || R_L) \approx R_L$, but let's compute the exact value:

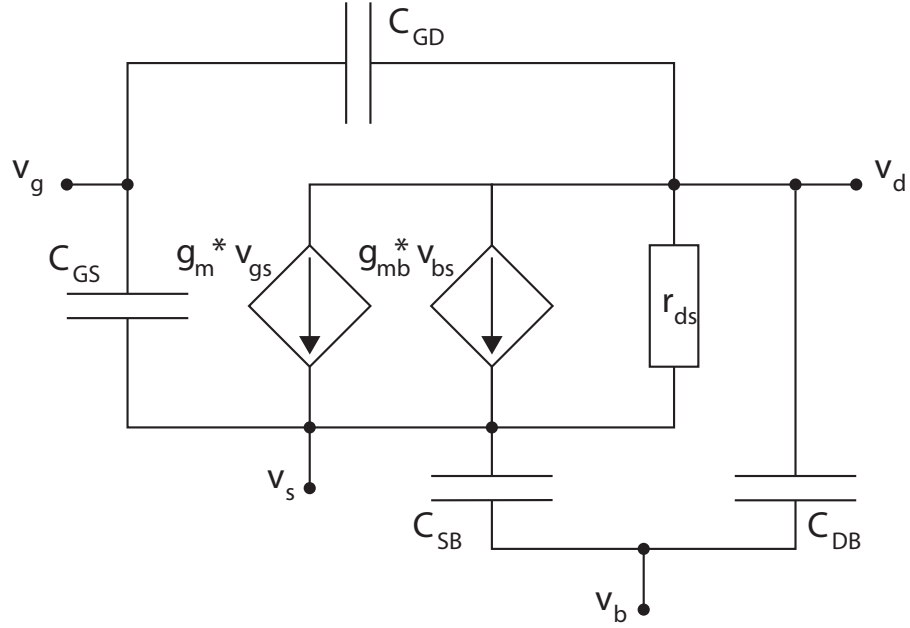


Figure 3.8: FET small signal equivalent for high frequency analysis.

$$(r_{ds} || R_L) = 9.23\text{M}\Omega \quad (3.23)$$

and thus:

$$A = 10.2 \quad (3.24)$$

3.5 FET high frequency small signal models

The capacitors inherent in the FET layout can be glimpsed in the cross sections of figures 1.1, 1.2, and 1.3. Whenever a conductor is either separated by silicon-oxide or a depletion region of a reverse-biased PN junction, we get a capacitor. In the small signal model, capacitors represent themselves and can simply be 'imported' from the large signal representation. These are C_{GS} , C_{GD} , C_{DB} , C_{SB} (see figure 3.8). The value of the capacitors C_{GS} and C_{GD} between the gate and the underlying conductor(s) depends on the oxide capacitance per area C_{OX} . C_{GS} scales with C_{OX} and the gate *area*, since it includes (most of) the channel underneath the gate. C_{GD} scales only with C_{OX} and the transistor *width* W , as it is caused by a fixed length of overlap of the gate onto the drain area. It's generally smaller than the other capacitors mentioned here, but often has an over-proportional influence (see section 9.4.5, buzzword 'Miller effect'). Capacitances of PN junctions depend on the doping profile and bias voltage, but are often considered approximately constant when the junction is reverse-biased, i.e. voltage independent. Then they will only depend on relative doping difference and are generally bigger for higher doping concentration differences so we can

give a characteristic capacitance per area for the drain and source regions C_J (the junction capacitance per area). In bulk CMOS these are capacitances C_{SB} and C_{DB} . In SOI CMOS, much the same way, we can also give a capacitance per area for source and drain C_{BOX} . Thus, source and Drain capacitors C_{SB} and C_{DB} scale with the source and drain area and either C_{BOX} or C_{PN} . Generally the source/drain lengths are left constant, so practically speaking they often only scale with transistor width W as well.

The capacitors play a role in the frequency dependent behaviour of analog circuits introduced in section 9.4 and speed and gate delays of digital circuits as discussed in section 4.2.

Chapter 4

FET Switch Models

In binary digital electronics, Boolean variables are represented by **binary** voltages¹, i.e. ideally the voltage signals will only have *two* discrete levels and either be at Gnd or Vdd. More realistically voltages are considered 'high', i.e. corresponding to logic 'true' or '1' or 'H', when they are above a certain minimum threshold V_{Hmin} , and low (i.e. logic 'false' or '0' or 'L') when they are below another threshold V_{Lmax} . The two thresholds are usually distinct, leaving a middle range where a signal cannot be guaranteed to be correctly identified as either 'high' or 'low'. Under nominal operating conditions, signals in a digital circuit should never settle in this 'undefined' state. However, while in a transitional state there are always time periods where one cannot rely on signal states.

Digital transistor models are voltage controlled **switches**, i.e. circuits with two conductive states between two terminals (S and D) determined by a switching threshold of the input control signal(s) on an (third) input terminal (that being G, compare figures 4.1 and 4.2). The two conductive states may not be all or none, but rather one state with high resistance and the other with low resistance.

Their most common use of such a switch in digital circuits when it is turned on is to either **pull down** a binary signal from Vdd to Gnd and keep it locked at Gnd thereafter, or vice versa: **pull up** a signal from Gnd to Vdd and lock it there. The former is implemented efficiently by nFETs (Fig. 4.1 B)) and the latter by pFETs (Fig. 4.1 C)). We'll discuss later why each of them is best suited for one of the two scenarios. Sometimes, if the switch is used for both, pull up *and* pull down, one can use a nFET and a pFET in parallel with inverted control signals (Fig. 4.1 D), aka **transmission gate**), for good switching performance. This is in particular used for switches between analogue signals, i.e. signals that are not just either Vdd or Gnd, but anything inbetween.

But for now let us discuss nFETs for pull down and pFETs for pull up actions. If we consider the limited conductance and generally parasitic capacitances of transistors and interconnect lines, the transition of such (digital) signals being pulled is not instantaneous and can be approximated with an RC-low pass filter or a constant current charging a load capacitance. These models are adapted to model the **propagation delay** t_p that causes delay in digital processing hardware and limits the speed of digital electronics. Assuming ideal switches with

¹and less commonly, currents

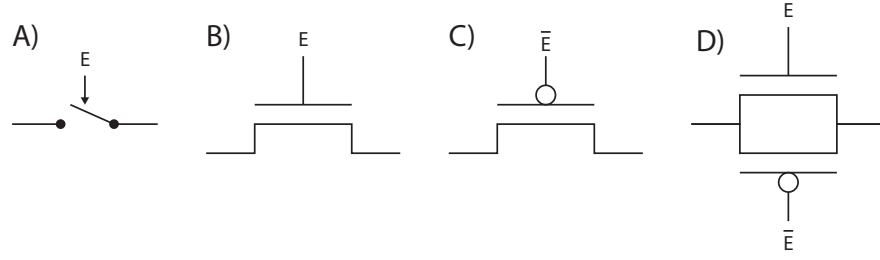


Figure 4.1: A) symbol for a switch. B) and C) single transistor implementations with severe non-ideality that a pFET is good for pulling up and an nFET for pulling down, but not vice versa. So if an nFET is used for pulling up, it will be much slower than pulling down. If this is not acceptable, one can use a **transmission gate** as depicted in D), i.e. both a pFET *and* an nFET in parallel with inverted control signals at their gate terminals. So at least one of them will always be a good conductor, irrespective of the voltages at the S/D terminals.

zero resistance or infinite current and/or no capacitance one can also disregard the delay and use the models to simply verify the correct implementation of logical functions.

The two states are 'switched' between dependent on the gate voltage V_G . Again, ideally this input/gate voltage would also be perfectly discontinuous (i.e. instant transition) and binary (always precisely at either Gnd or Vdd). However, as V_G itself may undergo a non-instantaneous transition, the idealized binary state of the switch is modelled as being governed by a hard threshold V_{sw} : an nFET is considered non-conducting (or having a large resistance R_{OFF} or a very small current I_{leak}) if $V_G < V_{sw}$ and conducting (with resistance R_{ON} that can be considered being zero or very small, or sourcing I_{ON}) if $V_G > V_{sw}$. See figure 4.2! Be aware that the switching threshold V_{sw} is normally NOT defined to be the same as the large signal threshold V_t ! It's often the voltage at which the balance of currents between a pFET and an nFET that form an inverter (see chapter 7) is shifting to either pull an output towards Vdd/high or Gnd/low. More simply, it is sometimes approximated as being exactly in the middle of the supply rails Vdd and Gnd:=0V, i.e. $V_{sw} = \frac{V_{dd}}{2}$. In the following we will adapt the latter, i.e. $V_{sw} = \frac{V_{dd}}{2}$.

The load capacitance C_L will include the same relevant parasitic capacitances as the small signal model presented in figure 3.8 (i.e. capacitors connected to the output/drain C_{GD}, C_{DB} of the stage under consideration and capacitors connected to the gate C_{GS}, C_{GD} of the next stage it is connected to) that are not explicitly shown in figure 4.2, as well as other 'parasitic' capacitances, e.g. of interconnecting metal traces. This modelling of the output allows to approximately determine signal propagation delays, i.e. the time the output needs to reach the switching threshold of the next stage in a digital signal processing chain.

The magnitudes of R_{ON} or I_{ON} are derived from a more complete transistor model or otherwise approximated quite crudely. The predictions for absolute propagation delay t_p will thus also be very crude. They shall, however, be

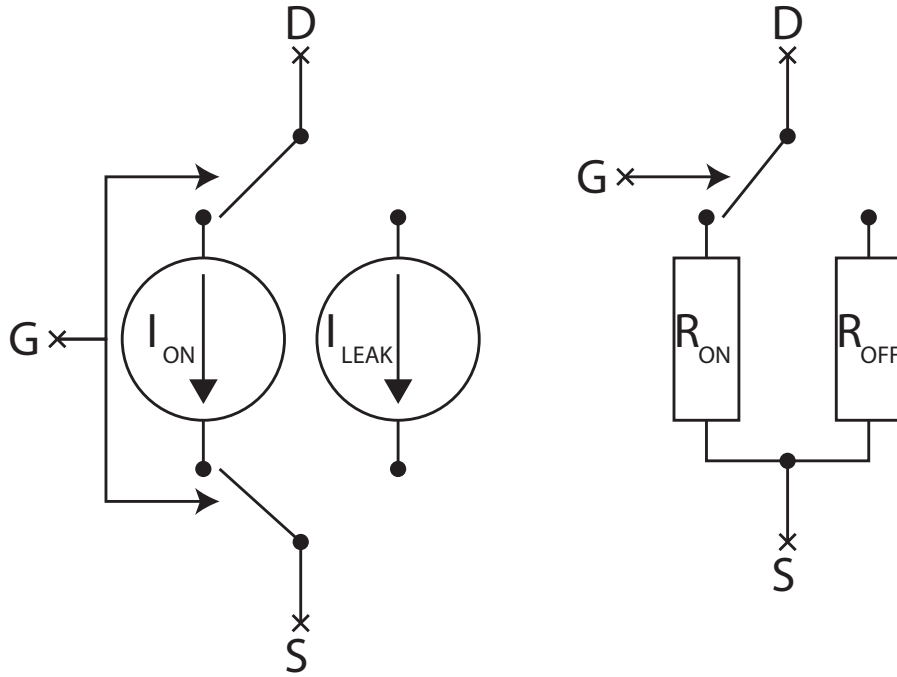


Figure 4.2: FET switch models. For an nFET the control terminal voltage $V_G > V_{sw}$ ($V_G < V_{sw}$ for pFETs) shorts the switch to R_{ON}/I_{ON} and $V_G < V_{sw}$ ($V_G > V_{sw}$ for pFETs) to R_{OFF}/I_{LEAK} . Often R_{OFF} and I_{LEAK} are neglected, i.e. considered to be $R_{OFF} = \infty\Omega$ and $I_{LEAK} = 0A$. For logic gates where the input is shorted to gate terminals of both nFETs and pFETs, V_{sw} is often defined as exactly in the middle of the supplies, i.e. when the supplies are Vdd and Gnd:=0V as $V_{sw} := \frac{V_{dd}}{2}$. However, realistically the input of that logic gate at which the output switches will always have some offset from $\frac{V_{dd}}{2}$.

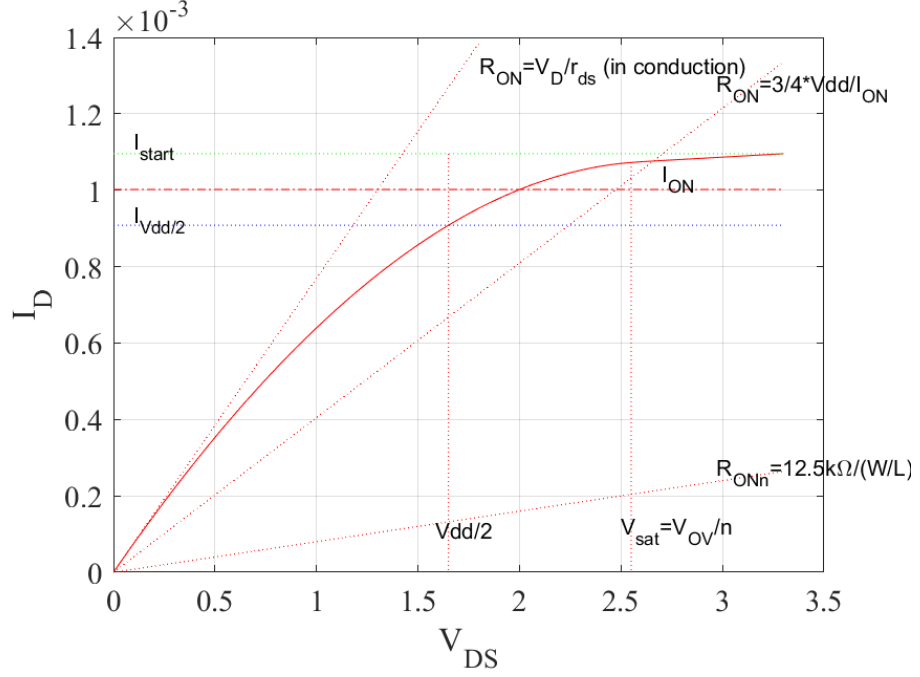


Figure 4.3: Comparing the large signal FET model with switch models. The drain current of an nFET according to the large signal model when its gate is at Vdd and its source at Gnd for parameters from an example 180nm CMOS process is shown as drawn out line. $R_{ON} = \frac{V_D}{I_{DS}}$ marks the approximation for R_{ON} appropriate to model the output resistance keeping the drain voltage V_D low, i.e. *after* a transition. If used to model the transition from high to low, the current is substantially overestimated, especially at the start of a transition. I_{ON} shows the constant current approximation where I_{ON} is the average current of the starting current and the current when the drain reaches $\frac{V_{dd}}{2}$. $R_{ONn} = 12.5\text{k}\Omega / (W/L)$ is the crude approximation from [2]. $R_{ON} = 3/4 * V_{dd}/I_{ON}$ is a more precise approximation based on I_{ON} .

very useful to predict *relative* propagation delays, i.e. in predicting the effect of changing circuit parameters such as $\frac{W}{L}$ in the design or choosing different CMOS technologies with different parameters μC_{OX} , or of reducing or increasing the power supply voltage Vdd.

In the following we will derive approximations of R_{ON} and I_{ON} under the assumption that we use nFETs as switches pulling a signal down towards Gnd, and pFETs as switches pulling a signal up towards Vdd. This assumption assures a good performance as the gate to source voltage difference will be constant and equal to a full Vdd for the entire transition, and thus the transistor's current in the large signal model will follow the I_D vs V_{DS} curve for a constant V_{GS} and be much larger than it would be if not only V_{DS} but also V_{GS} would be shrinking under the transition.

4.1 Static output resistance R_{ON}

If one considers that generally the threshold voltage is a fraction smaller than $\frac{1}{2}$ of the supply voltage Vdd, one notices that an nFET with V_S at Gnd and Vdd as its gate voltage V_G is in conduction for most of the range of drain voltages $V_D \in [0, \text{Vdd}]$ (see figure 4.3). So one might approximate R_{ON} with the equation valid for deep conduction where the transistor actually DOES behave like a resistors, i.e. equations (2.23) and (2.25), for respectively strong- and weak inversion. The curve for $R_{ON} = r_{ds}$ is also plotted for comparison. This is a good approximation for R_{ON} at the end of a transition when the transistor is *maintaining* a binary state, i.e. a measure of how much resistive load can be overcome to reach and maintain a voltage close to the supply rails. At this instance, please note that this is NOT the case if you use a pFET as a pull down device or an nFET as a pull up device, because in THAT case you will attempt to pull the source terminal voltage V_S and NOT the drain terminal voltage V_D . In THAT case, at the end of a transition you will not only have a $V_{DS} = 0$ but also $V_{GS} = 0$ and thus an extremely high resistance/low conductance trying to tie that V_S to respectively Gnd and Vdd, and extremely low capability of overcoming any kind of resistive load at that terminal pulling the voltage elsewhere!

So just to repeat: equations (2.23) and (2.25) provide good approximations for R_{ON} to gauge the pull down capability of nFETs and the pull up capability of pFETs to *maintain* a digital state.

4.2 Dynamic transitions and propagation delay

4.2.1 Approximating a constant I_{ON} for signal transition

For modelling an entire transition of V_D from one digital/logic state to the switching point, however, (2.23) and (2.25) do overestimate the actual current, especially if the threshold voltage is relatively high compared to the supply voltage Vdd (see figure 4.3). In that case the device will be in saturation rather than conduction for a larger part of the transition and will 'cap' the current at a maximum level for large V_{DS} .

[2] suggests to derive I_{ON} as probably the best approximative value so far from the large signal model as the average of the drain currents at the start of the transition I_{start} , i.e. from an nFET pulling down when $V_D = \text{Vdd}$, and the drain current $I_{\frac{\text{Vdd}}{2}}$ when the switching threshold is reached, i.e. $V_D = \frac{\text{Vdd}}{2}$, thus reaching the switching point and flipping a next switch connected to that drain.

$$I_{ON} = \frac{I_{\text{start}} + I_{\frac{\text{Vdd}}{2}}}{2} \quad (4.1)$$

For the best result one uses the full EKV model to compute those two currents. This is also indicated in figure 4.3. Note that in any case this current will scale with $k = \mu C_{OX} \frac{W}{L}$, so the primary weapon of the designer to change propagation delay in a given CMOS technology node is to adjust $\frac{W}{L}$.

The book [2] derived the two currents I_{start} and $I_{\frac{\text{Vdd}}{2}}$ for the very specific (and traditionally the most common) case of strong inversion and with the region operation approximations, yielding a somewhat less exact result. However,

beyond the influence of parameter k it will allow us to express an important relationship of t_p and Vdd as well as the threshold voltage V_t relative to Vdd.

So for I_{start} one can use the appropriate saturation region equation. If one uses the strong inversion approximation (assuming $n = 1$, $\lambda = 0$ and $V_S = 0V$) one gets:

$$I_{\text{start}} = \frac{1}{2}k(V_{\text{dd}} - V_t)^2 \quad (4.2)$$

For $I_{\frac{V_{\text{dd}}}{2}}$ one uses the appropriate conduction expression, e.g. for strong inversion, $n = 1$ and $V_S = 0V$, and on the condition that we are in conduction, i.e. $\frac{V_{\text{dd}}}{2} = V_D < V_{OV} = V_{\text{dd}} - V_t \Rightarrow V_t < \frac{V_{\text{dd}}}{2}$:

$$I_{\frac{V_{\text{dd}}}{2}} = k\frac{1}{2}V_{\text{dd}} \left[\frac{3}{4}V_{\text{dd}} - V_t \right] \quad (4.3)$$

(Once more: Note that if you happen to be in weak inversion, that you have to compute the two currents appropriately with those equations, i.e not the ones used here for strong inversion!)

Thus, for strong inversion we can derive a more direct solution for the average of the two:

$$I_{ON} = \frac{1}{4}k \left((V_{\text{dd}} - V_t)^2 + \frac{3}{4}V_{\text{dd}}^2 - V_{\text{dd}}V_t \right) \quad (4.4)$$

$$= \frac{1}{4}k \left(V_{\text{dd}}^2 - 2V_{\text{dd}}V_t + V_t^2 + \frac{3}{4}V_{\text{dd}}^2 - V_{\text{dd}}V_t \right) \quad (4.5)$$

$$= \frac{1}{4}k \left(\frac{7}{4}V_{\text{dd}}^2 - 3V_{\text{dd}}V_t + V_t^2 \right) \quad (4.6)$$

$$= \frac{1}{4}kV_{\text{dd}}^2 \left(\frac{7}{4} - 3\frac{V_t}{V_{\text{dd}}} + \frac{V_t^2}{V_{\text{dd}}^2} \right) \quad (4.7)$$

$$= \frac{1}{4}kV_{\text{dd}}^2 \left[\left(\frac{3}{2} - \frac{V_t}{V_{\text{dd}}} \right)^2 - \frac{1}{2} \right] \quad (4.8)$$

With the condition from above that $V_t < \frac{V_{\text{dd}}}{2}$, we get an increase of this current with decreasing relative threshold $\frac{V_t}{V_{\text{dd}}}$, i.e. a faster switching at the cost of more current and thus power. Consequently, when lowering Vdd, this current is reduced quadratically with the term V_{dd}^2 as well as with the slope of the term $\left(\frac{3}{2} - \frac{V_t}{V_{\text{dd}}} \right)^2$.

Also note that the mobility (part of parameter $k = \mu C_{OX} \frac{W}{L}$) for pFETs is generally lower in P-bulk processes (which are the most common kind). So for same size transistors one usually gets a bigger current and faster switching for nFETs than for pFETs. If symmetry is important, one can compensate for that by increasing $\frac{W_P}{L_P}$ for pFETs in a logic circuit. More of that later in section 7.1.1!

So once a switch input V_G at the gate G reaches V_{sw} and the switch in the models in Fig. 4.4 is assumed to be shorted, one measures the **propagation delay** t_p as the time it takes from when the switch is flipped until the output reaches the switching threshold V_{sw} of the next stage. For now we assume instant switching so the output starts immediately to move from its resting potential

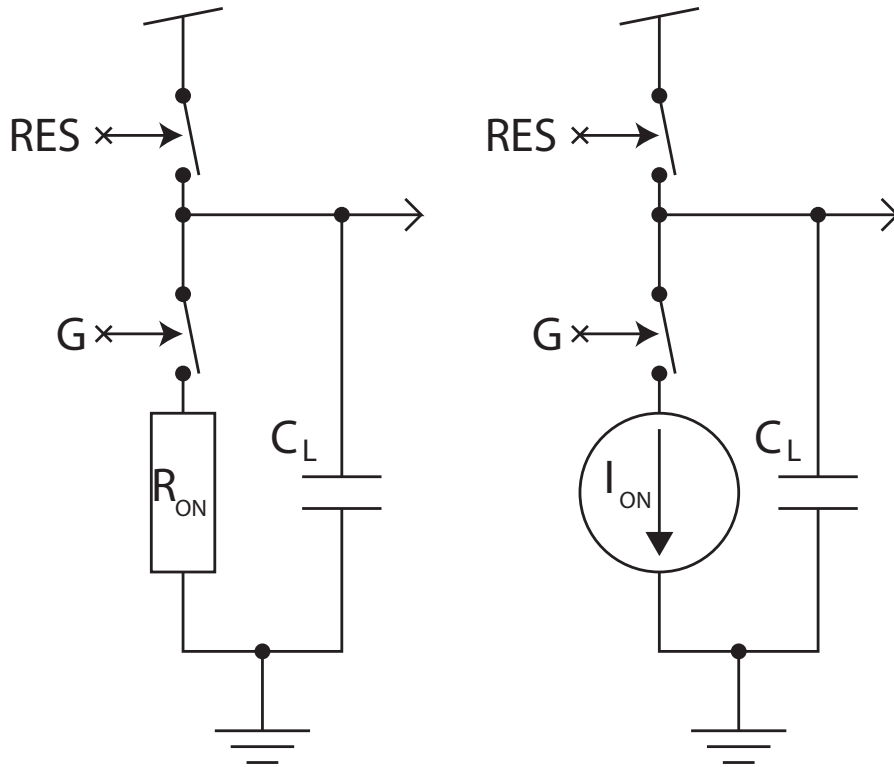


Figure 4.4: Illustrating the scenario to measure propagation delay in case an nFET is used as a switch to pull down its drain voltage from V_{dd} to $\frac{V_{dd}}{2}$. The reset switch RES is added here to reset the output to its starting voltage V_{dd} and establish the right starting point.

V_{start} at one of the two supply rails. However, if one wants to determine a propagation delay if the switch is modelled more realistically or in a physical measurement and the switch control signal G is not changing binary states infinitely fast, one defines the starting time for the propagation delay as the moment in which the control signal G of the switch reaches a defined switching threshold V_{sw} for *this* switch. (Often the simplifying assumption is made that $V_{sw} = \frac{V_{dd}}{2}$)

This is illustrated in figure 4.4 for an nFET pulling its drain voltage low. In case of a current source the voltage drops linearly and the time to reach $V_{sw} := \frac{V_{dd}}{2}$ starting from Vdd is then:

$$t_p = \frac{V_{dd}C_L}{2I_{ON}} \quad (4.9)$$

$$= \frac{V_{dd}C_L}{2\frac{1}{4}kV_{dd}^2 \left[\left(\frac{3}{2} - \frac{V_t}{V_{dd}} \right)^2 - \frac{1}{2} \right]} \quad (4.10)$$

$$= \frac{2C_L}{kV_{dd} \left[\left(\frac{3}{2} - \frac{V_t}{V_{dd}} \right)^2 - \frac{1}{2} \right]} \quad (4.11)$$

4.2.2 Approximating a constant R_{ON} for signal transition

A resistive model instead of a current source model can be preferable (i.e. is easier to use), when assembling more complicated digital gates that will require transistors in series (see section 7.1.1). The Sedra/Smith book [2] suggests the crudest approximation yet based on analysis of CMOS technology nodes from $0.25\mu\text{m}$ down to 180nm , where (very!) approximately all had the same resistance per $\frac{W}{L}$ ratio for nFETs and pFETs respectively:

$$\begin{aligned} R_{ON_n} &= \frac{12.5}{\frac{W_n}{L_n}} \text{k}\Omega \\ R_{ON_p} &= \frac{30}{\frac{W_p}{L_p}} \text{k}\Omega \end{aligned} \quad (4.12)$$

The use of this model is purely to gauge the relative effects of changing $\frac{W}{L}$ and has no predictive value with respect to changing Vdd or the CMOS technology, nor for the absolute value of the propagation delay t_p . In figure 4.3 this approximation is indicated near the real curve, and would actually substantially underestimate the real drain current for this particular CMOS process.

A more accurate approximation in absolute terms one can use the current model for I_{ON} from equation (4.1) and derive a resistance that would result in the same average current:

$$\begin{aligned}
\frac{\frac{V_{dd}}{R_{ON}} + \frac{\frac{V_{dd}}{2}}{R_{ON}}}{2} &= I_{ON} \\
\frac{3}{4} \frac{V_{dd}}{R_{ON}} &= I_{ON} \\
R_{ON} &= \frac{3}{4} \frac{V_{dd}}{I_{ON}}
\end{aligned} \tag{4.13}$$

And in case of a resistive model with a load capacitance on the drain one gets exponential decay of the output voltage and the resulting expression:

$$\begin{aligned}
e^{\frac{-t_p}{R_{ON}C_L}} &= \frac{1}{2} \\
t_p &= -R_{ON}C_L \ln\left(\frac{1}{2}\right) \\
t_p &= 0.69R_{ON}C_L
\end{aligned} \tag{4.14}$$

4.2.3 What's next

In part III of this book, these switches are combined to **combinational logic circuits** implementing **Boolean functions** and **sequential logic circuits** implementing **finite state machines (FSM)**.

Part II

Linear Circuit Analysis Basics

Chapter 5

Linear circuit filters

5.1 Linear filter and transfer function

Definition: A **linear system** is an operator h that transforms an input signal (i.e. a function $x(t)$ of time t) into an output signal $h(x(t))$ such that $h(x(t)) + h(y(t)) = h(x(t) + y(t))$ and $sh(x(t)) = h(sx(t))$, for any scalar value s .

We also call h a **linear filter**. Incidentally any electronic circuit composed of linear circuit elements fulfils the requirement of a linear system, i.e. is a linear filter. (No proof is presented here, but believe you me!) Low level linear circuit elements are real electronic passive components, i.e. resistors, capacitors and inductors, and more artificial active elements such as ideal current and voltage sources that are either constant or linearly dependent on a signal in the circuit (e.g. transconductances, transresistances, voltage- and current-amplifiers). So when replacing transistors with their small signal equivalent, linear circuit analysis then becomes a major tool in analysing transistor circuits. That is the reason why this topic occurs here.

In the following it is assumed that the reader is familiar with the Kirchhoff's law and the Ohm's law. They are just briefly repeated here without further explanations or illustrations. Some examples of how to do circuit analysis with them will follow.

Kirchhoff's current law: the sum of all currents into an electric node is zero.

Kirchhoff's voltage law: the sum of all voltages along a electric circuit loop is zero.

Ohm's law: current I through resistor R and voltage V across resistor R are governed by the rule

$$V = I * R \text{ (or later } V = I * Z) \quad (5.1)$$

Where R is a **resistance** (actually defined by the having the property described by Ohm's law) and Z is an **impedance** (which includes capacitors and inductors in addition to resistances, as well as networks of these three elements). The symbols used in this compendium for passive linear impedances are shown in figure 5.1.

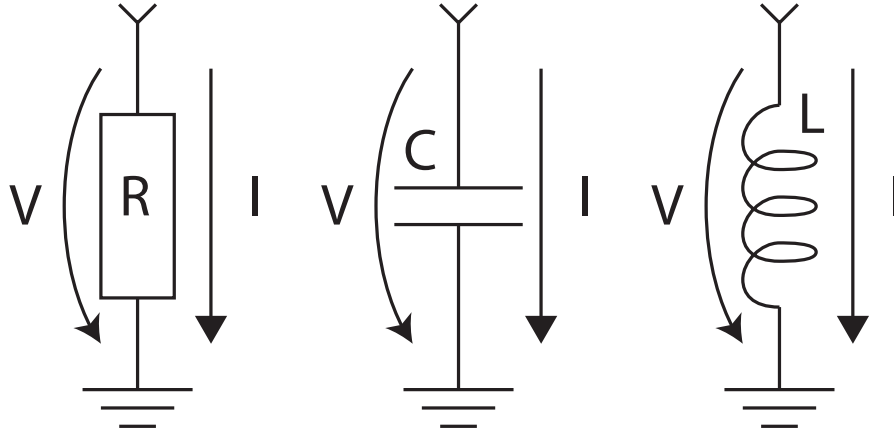


Figure 5.1: Passive linear circuit elements symbols, i.e. a resistor, a capacitor, and an inductor. See also table 5.1

Linear circuit analysis uses Ohm's and Kirchhoff's laws to find the currents and voltages in a network of interconnected circuit elements. This is straight forward for operations that only use momentary signal values as inputs, e.g. operations introduced by using resistors and linearly dependent sources (i.e. not impedances including capacitors and inductors). Obviously any relationship between momentary (i.e. no dependence on history) current- and voltage values in a network consisting of only such elements can be expressed as a linear combination of other momentary signal values in the network. In other words $h(x(t))$ will then simply be a linear **function**, mapping scalar x to a scalar $h(x)$ where the history of the input signal ($x(t)$) does not affect the result in any way (as opposed to an **operator**, mapping function $x(t)$ of time to a function $h(x(t))$ of time, where $h(t)$ may depend on the entire signal history of $x(t)$).

However, inductances L and capacitances C introduce operations using the derivative or integral of some signals (in other words the results will depend on signal history/dynamic) (also compare table 5.1):

$$I = CV \frac{\partial}{\partial t} \quad (5.2)$$

$$V = LI \frac{\partial}{\partial t} \quad (5.3)$$

Three principles are used to tackle such differential operators in linear circuit analysis without actually having to resort to solving differential equations:

1. Any signal can be decomposed into its frequency components, i.e. into a sum/integral of sine waves at different phases and amplitudes. This decomposition is known as the **Fourier Transform**.
2. The defining property of linear systems allows for an input signal to be decomposed into sine waves that can be sent through the system individually and reassembled/summed up at the output.

3. Linear combinations of different sine waves of the same frequency but different phases will always result in a sine wave of the same frequency (with its own phase and amplitude).

So if we do know how a linear filter h treats a *sine wave* for any particular frequency and phase, we have completely described that filter and know what it does to any signal $x(t)$. Furthermore what it DOES do to a sine wave input is change its amplitude and shift its phase, but leaves the frequency untouched. We can then derive a **transfer function** $H(\omega)$ of frequency (instead of of time like the filter $h(t)$) that describes the change of amplitude and phase for each frequency given an input signal frequency spectrum $X(\omega)$ (the Fourier transform of $x(t)$).

So if in an electronic circuit there is a single driving input node receiving a pure sine wave of frequency ω and phase φ , then any other signals in the network should be sine waves of the same frequency. Let us verify this also for those circuit elements introducing differential operators. For instance the current I through a capacitor of capacitance C when the voltage V across it is defined as

$$V := \sin(\omega t + \varphi) \quad (5.4)$$

is given by

$$I = CV \frac{\partial}{\partial t} = C\omega \cos(\omega t + \varphi) = C\omega \sin(\omega t + \varphi + \frac{\pi}{2}) \quad (5.5)$$

So a capacitor will cause a current through it to be a phase shifted copy with different magnitude of a sine wave voltage across it, where the phase $\angle I$ is given by the phase $\angle V$ of the voltage and a $\frac{\pi}{2}$ phase shift.

$$\angle I = \angle V + \frac{\pi}{2} \quad (5.6)$$

and the magnitude $|I|$ increases with frequency (i.e. a capacitor acts more and more like a short at higher frequencies).

$$|I| = C\omega|V| \quad (5.7)$$

Other nodes in our network h of circuit elements with a single sine wave input may now be linear combinations of phase shifted signals and/or may introduce more phase shift. So take note that any linear combination of sine waves of the same frequency ω but different magnitude and phase, will again result in a sine wave of the **same frequency** but different magnitude and phase. Figure 5.2 illustrates that geometrically, using the fact that the projection of a vector onto the y-axis is the sine of the angle of the vector's polar representation multiplied with the vector length. So we assume two sine waves $a \sin(\omega t)$ and $b \sin(\omega t + \varphi)$ with a relative phase shift φ . If you want to compute the resulting sine wave $c \sin(\omega t + \gamma)$ when summing those two sine waves you can do a vector summation of the polar representation vectors with their angle being the argument of the sine function and their length being the sine waves' amplitudes. The resulting vector's length is c and its polar form angle is γ . From figure 5.3 showing the situation at $t = 0$ we can glean the geometrical relationship:

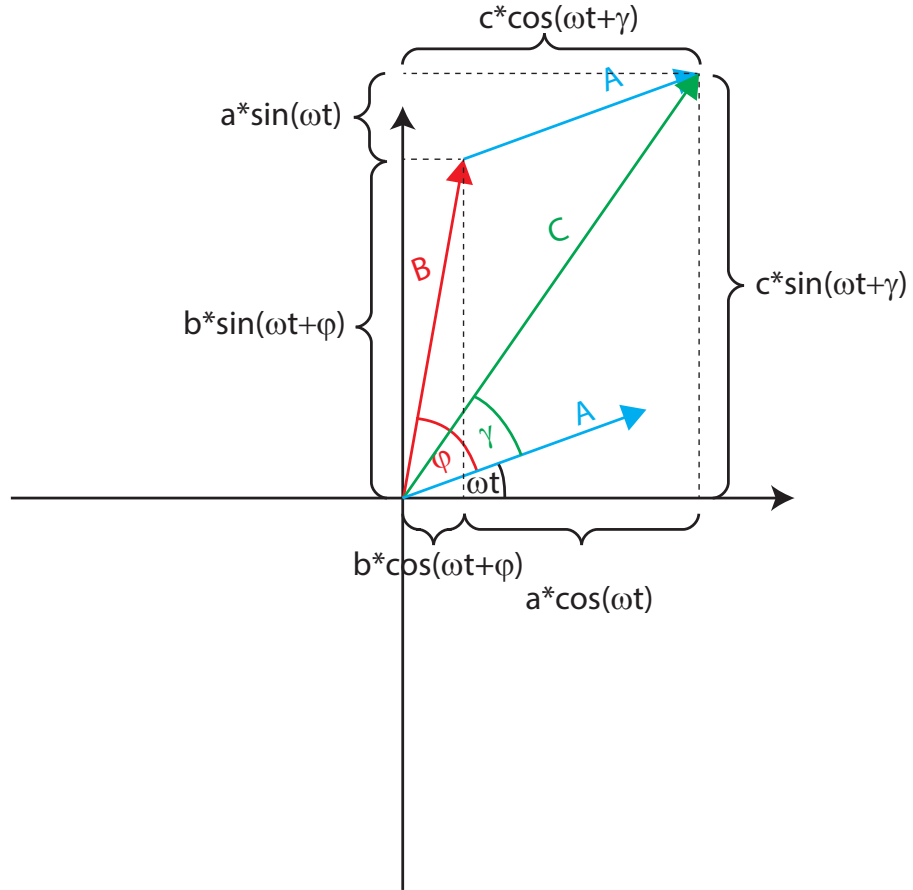


Figure 5.2: Geometrical illustration of the fact that a linear combination of two sine waves of the same frequency will result in a sine wave of the same frequency at different phase and magnitude. (This is ultimately the explanation why complex number arithmetic is so convenient to describe transfer functions, as will be shown later.)

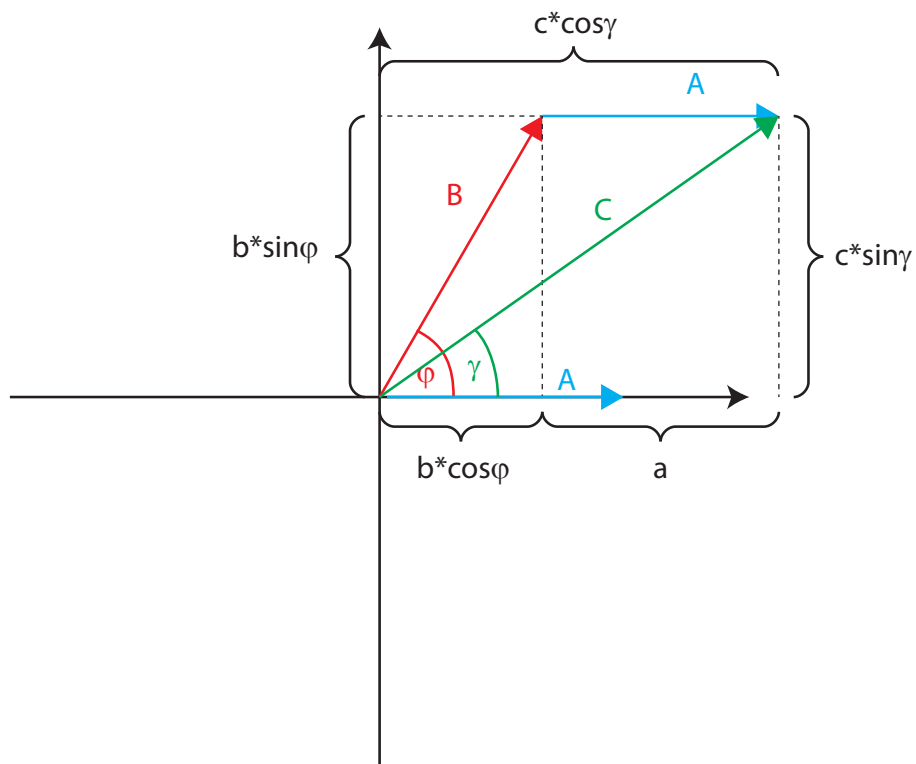


Figure 5.3: The same illustration as in figure 5.2 at $t = 0$. It can conveniently be used to derive c and γ as given in equation (5.8).

	time-space	frequency-space ($s := j\omega$)		
	Ohm's law	impedance	admittance	Ohm's law
resistor	$V = IR$	R	$\frac{1}{R}$	$V = IR$
capacitor	$I = CV \frac{\partial}{\partial t}$	$\frac{1}{sC}$	sC	$I = VsC$
inductor	$V = LI \frac{\partial}{\partial t}$	sL	$\frac{1}{sL}$	$V = IsL$

Table 5.1: A table of passive circuit elements we'll encounter in this compendium in linear electronic systems.

$$\begin{aligned}
 a \sin(\omega t) + b \sin(\omega t + \varphi) &= c \sin(\omega t + \gamma) \\
 c &= \sqrt{(b \sin(\varphi))^2 + (b \cos(\varphi) + a)^2} \\
 \gamma &= \tan^{-1} \left(\frac{b \sin(\varphi)}{b \cos(\varphi) + a} \right)
 \end{aligned} \tag{5.8}$$

Coincidentally (or not so coincidentally) this geometrical analysis can nicely be mapped to complex number representation in the (Re,Im) plain using polar form. So it happens to be the case that one can actually use complex number arithmetic to express the exact same relation: let A and B be complex numbers which are expressed in polar form $A = ae^{j\omega t}$ and $B = be^{j\omega t + \varphi}$. Then it so happens that

$$A + B = ae^{j\omega t} + be^{j\omega t + \varphi} = ce^{j\omega t + \gamma} \tag{5.9}$$

Where c and γ comply with (5.8).

Also, the relation (5.5) can actually be achieved when sine waves I and V are represented as complex numbers where $I = |I|e^{j\omega t + \angle I}$ and . Then the relation (5.5) can simply be expressed as:

$$I = e^{j\frac{\pi}{2}} \omega CV = j\omega CV \tag{5.10}$$

Thus, to deduce currents and voltages in a linear circuit network in frequency space one can use Kirchhoff's laws and Ohm's law just like with only resistors/conductors, but now using complex number arithmetic and any capacitor or inductor enters the equations as a frequency dependent imaginary component as indicated in table 5.1. Note that one refers to a resistance (generally using the letter R) as the constant that turns a current into a voltage if that constant has no imaginary part. If that constant may have an imaginary component one refers to it more generally as **impedance**, generally using the letter Z . Correspondingly a conductance (the inverse of a resistance, i.e the constant applied to a voltage to result in a current not containing an imaginary component, generally represented by the letter G) that may have an imaginary component is more generally referred to as **admittance**, generally represented by letter Y . (Do not get confused in the following where Y is often used for the output of a filter!)

So if a known linear electronic circuit is used as a linear filter $y(t) = h(x(t))$, i.e. one (or some) element terminal's current or voltage is considered the input of the filter and another (or others) element's terminal current or voltage the

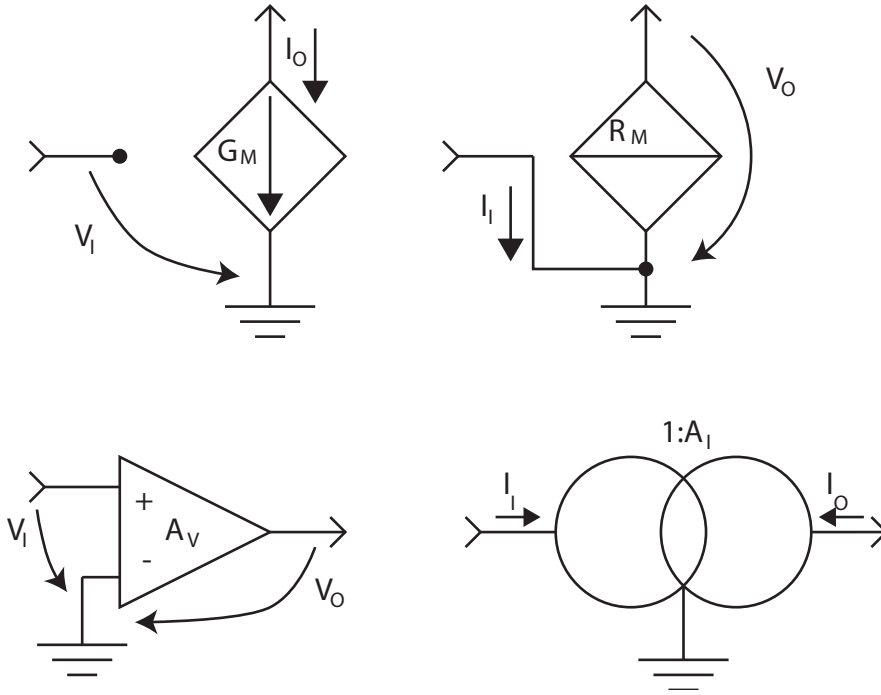


Figure 5.4: Ideal active linear circuit elements symbols, i.e. a transconductance, a transresistance, a voltage amplifier and current amplifier. See also table 5.2.

transconductance	$I_O = G_M V_I$
transresistance	$V_O = R_M I_I$
voltage amp	$V_O = A_V V_I$
current amp	$I_O = A_I I_I$

Table 5.2: A table of ideal active circuit elements we'll encounter in this compendium in linear electronic systems. These are three terminal devices where the input signal is applied to a different terminal than the output signal, both referenced to a common Gnd terminal. The output is then provided from an ideal current or voltage source. The current source in the transistor small signal model is an example (see 3.4).

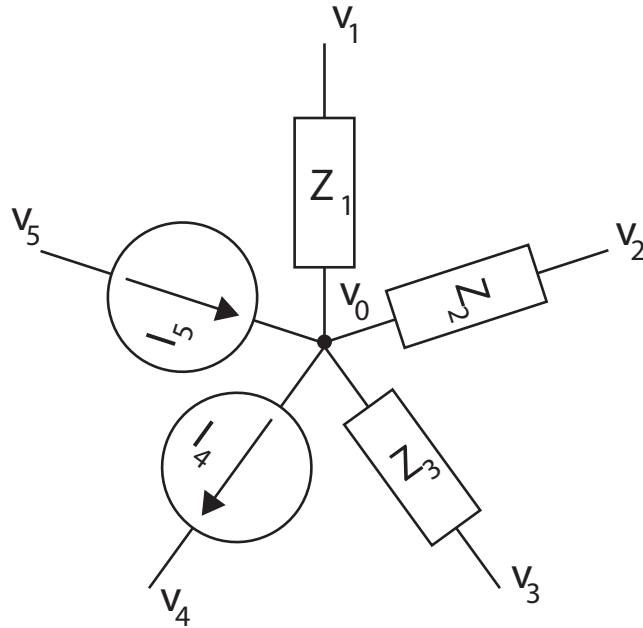


Figure 5.5: Example of a node in a circuit to illustrate a method to quickly write down the equation resulting from Kirchhoff's current law for node v_0 .

output, one can express the input-output relationship as the output equal to the input multiplied with a linear combination of all the impedances/admittances of the network. So the relationship $y(t) = h(x(t))$ expressed as differential equation in the time space becomes a simple multiplication in frequency space $Y(\omega) = H(\omega) * X(\omega)$. For each frequency ω , $H(\omega)$ will be a complex number $H(\omega) = |H(\omega)| * e^{j\angle H}$ and the multiplication with it will cause a multiplication of the amplitude of the input with $|H|$ and a phase shift $\angle H$.

$$|Y| = |H||X| \quad (5.11)$$

$$\angle Y = \angle H + \angle X \quad (5.12)$$

5.2 Deriving a transfer function from a linear circuit

Using Kirchhoff to get an I/O relationship of a linear circuit is done defining an input signal x and an output signal y (those can be voltages or currents) and setting up a set of equations in frequency space for the circuit voltage nodes and currents through the circuit elements. One can then solve the set of equations eliminating all currents and voltages other than x and y . Thus one can then write down the transfer function as $h = \frac{y}{x}$

Setting up enough independent equations to eliminate all unknowns can be done using Kirchhoff's current or voltage laws or a combination thereof. One

systematic approach is to use the current law in each circuit node. Doing it always the same way can make it more difficult to commit mistakes, so consider always to use the following approach. Have a look at the example of one node as it might occur in your circuit with all its connected nodes in figure 5.5. The method is to write down the equation for each node with unknown voltage (so no need to do it for the input if it is a voltage (!), as this will be defined as the starting signal upon which the output depends), by adding on the left hand side of the equation the voltage of the node in question, here it's v_0 , multiplied with all the admittances (i.e. the inverse of the impedances) connected to that node, as well as any explicit current source drawing current *out* from node v_0 . Then place on the right hand side the sum of all voltages appearing on the other end of the admittances multiplied with that admittance, as well as any explicit current sources supplying current *into* v_0 . So for this example we get:

$$v_0 \left(\frac{1}{Z_1} + \frac{1}{Z_2} + \frac{1}{Z_3} \right) + I_4 = v_1 \frac{1}{Z_1} + v_2 \frac{1}{Z_2} + v_3 \frac{1}{Z_3} + I_5 \quad (5.13)$$

So that's one equation and you'll get one for each node of the circuit with unknown voltage. Thus the set of equations shall be enough to eliminate those unknown voltages.

For an exhaustive example please have a look at the derivation of the transfer function of a general single stage transconductance amplifier in section 9.4.1.

5.3 Thévenin and Norton equivalent circuit, input- and output impedance

In the following we will often substitute entire networks of linear circuit elements between two connection points with a 'black box' characterized only by a total impedance seen between the contact points and for 'active' circuits with an additional signal-, biasing-, or power source in series. If this is a voltage source, the substituting equivalent circuit is called to be in **Thévenin** form or alternatively if the source is a current source in parallel it is in the **Norton** form. The two forms are equivalent and one can choose the one that is preferable in a given situation.

The Thévenin and Norton forms are for example useful as more realistic signal source models (e.g. a sensor signal) included into a electronic system model, or to simplify an entire subcircuit of a complicated system before further analysis by reducing that subcircuit to its Thévenin or Norton equivalent.

In general any 'black box' two terminal circuit of linear components (i.e. passive linear circuit elements and current/voltage sources that are either constant or linearly dependent on other signals) can be expressed as Thévenin or Norton equivalent circuit (Fig. 5.6), with an equivalent resistance R_S or more generally impedance Z_S and a signal or DC source. One often talks about the 'impedance/resistance seen in this node' where the second node implicitly is Gnd, and thus often of the **input impedance/resistance** and **output impedance/resistance** if that node happens to be the input or output of an electronics component in a system context.

To determine the resistance in this node, and thus find the Thévenin/Norton equivalent circuit, stop the time varying signal (if any and if possible) and supply

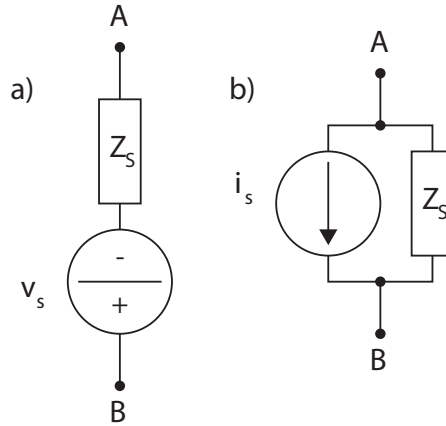


Figure 5.6: Two *equivalent* models of signal or power/biasing sources (e.g. a sensor or other *transducer*): a) is called the Thévenin form and b) the Norton form. Note: **Output impedance Z_S** is the same in both models while $v_s = i_s Z_S$. Oftentimes one considers it to be purely resistive, so you can write **R_S** instead of Z_S and one talks of an **output resistance**. You may also call the impedance/resistance more generally the 'impedance/resistance looking into the circuit from node A'. It might in fact also be an *input* node in the overall circuit context one is looking at.

a small test current i_X or test voltage v_X on top of the DC point of operation between the nodes and measure/compute the resulting change in voltage or current respectively. R_S is then $\frac{\delta v_X}{\delta i_X}$. To measure impedance, you do the same but using an AC signal swept across the frequency spectrum as test signal and measure the phase and magnitude of the resulting voltage or current respectively.

To get the DC level of the signal source, for the Thévenin equivalent, measure/compute the output voltage (voltage-signal) for open circuit between the two terminals, and for the Norton form, measure/compute the short circuit current (or current-signal) when node A is shorted to node B.

The Thévenin can be transformed into the Norton form or vice versa by using $v_s = Z_S i_s$.

5.4 Bode plot, poles and zeros

H can conveniently be described with a **Bode plot** where its magnitude and phase shift are plotted in two graphs vs frequency ω . See the example in figure 5.7! It shows the Bode plot of a so called low pass filter, where DC voltages and low frequency AC voltages 'pass' through with a constant amplification $A_{DC} := |H(0)| = 100$, but with $|H(\omega)|$ starting to decline at about $\omega = 10^5$ and decline more steeply at $\omega = 10^7$. This is typical for what is called a second order low pass filter. Also note (for later reference) that the phase shift is in the middle of a smooth step of a total of $-\frac{\pi}{2} = -90^\circ$ at those two frequencies. Note here that the ω axis and the $|H|$ axis usually are given in logarithmic scale to better represent $\frac{1}{\omega^2}$ hyperbolic decay. That's for low pass filters or also ω^i

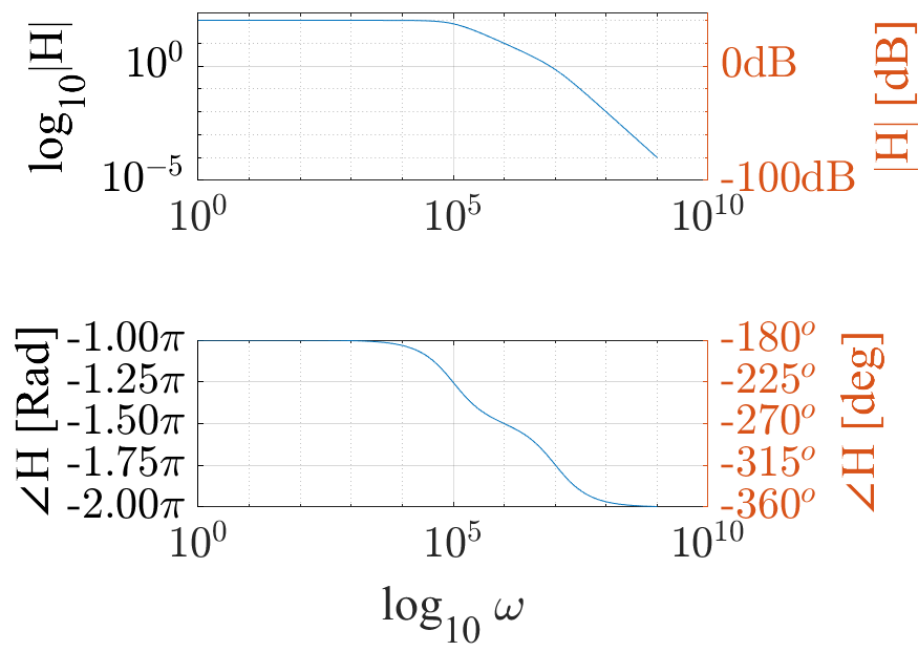


Figure 5.7: An example of a Bode plot of a second order low pass filter, with real poles at frequencies $\omega = 10^5$ and $\omega = 10^7$

hyperbolic growth for high pass or band pass filters, or in general ω^i where i can be positive or negative: a factor ω^i will result in a straight line with inclination i . In particular the logarithmic $|H|$ axis is usually drawn in Decibel (dB), for historical reasons and to annoy students (as well as this author), where 20dB correspond to a factor 10 in amplitude:

$$\begin{aligned} A_{\text{dB}} &= 20 \log_{10} A_{\text{lin}} \\ A_{\text{lin}} &= 10^{\frac{A_{\text{dB}}}{20}} \end{aligned} \quad (5.14)$$

In this example in figure 5.7 H is:

$$H = A_{DC} \frac{1}{(1 + \frac{s}{\omega_{p_1}})(1 + \frac{s}{\omega_{p_2}})} \quad (5.15)$$

Where $A_{DC} = -100$, $\omega_{p_1} = 10^5$, and $\omega_{p_2} = 10^7$. The phase shift here starts at $-\pi \text{Rad} = -180^\circ$, because A_{DC} is negative. So already at low frequencies the output has a -180° phase shift to begin with.

Important notice: all other literature out there chooses to start the Bode plot of both inverting and non-inverting low pass transfer functions with a **zero degrees** (0°) phase shift by convention and by long standing tradition. The author of this compendium chooses to break with this tradition since some of the subsequent explanations (in particular of phase margin in section 5.7.2) become easier (i.e. less confusing) that way, in his opinion. But be aware that most other texts will show Bode plots of inverting low pass filters having a 0° phase shift at DC!!!

5.4.1 Root form of the transfer function

In general, when solving a linear electronic network's I/O relationship, e.g. $v_o = H v_i$, H can be written as the ratio of two polygons in s , i.e.:

$$H = \frac{A}{B} = \frac{a_0 + a_1 s + a_2 s^2 \dots a_n s^n}{b_0 + b_1 s + b_2 s^2 \dots b_m s^m} \quad (5.16)$$

And for convenience this expression is usually manipulated into **root form** by bringing $\frac{a_0}{b_0}$ as a constant in front of the division, and then finding the solutions $\{z_1, \dots, z_n\}$ for $\frac{A}{a_0} = 0$, and $\{p_1, \dots, p_m\}$ for $\frac{B}{b_0} = 0$. We refer to these solutions as **poles** p_i and **zeros** z_i . Oftentimes in the following they will be real negative numbers. For convenience we sometimes refer to their magnitude as **pole frequencies** $\omega_{p_i} = |p_i|$ and the **zero frequencies** $\omega_{z_i} = |z_i|$. One can rewrite (5.16) as a product of first order terms:

$$H = \frac{a_0}{b_0} \frac{(1 + \frac{s}{-z_1}) \dots (1 + \frac{s}{-z_n})}{(1 + \frac{s}{-p_1}) \dots (1 + \frac{s}{-p_m})} \quad (5.17)$$

So if $s \in \{z_1, \dots, z_n\}$, H becomes zero ($H = 0$), and if $s \in \{p_1, \dots, p_m\}$ H becomes infinite ($H = \infty$), i.e. has a pole.

If the poles and frequencies are real and negative, (5.17) is the same as:

$$H = \frac{a_0}{b_0} \frac{(1 + \frac{s}{\omega_{z_1}}) \dots (1 + \frac{s}{\omega_{z_n}})}{(1 + \frac{s}{\omega_{p_1}}) \dots (1 + \frac{s}{\omega_{p_m}})} \quad (5.18)$$

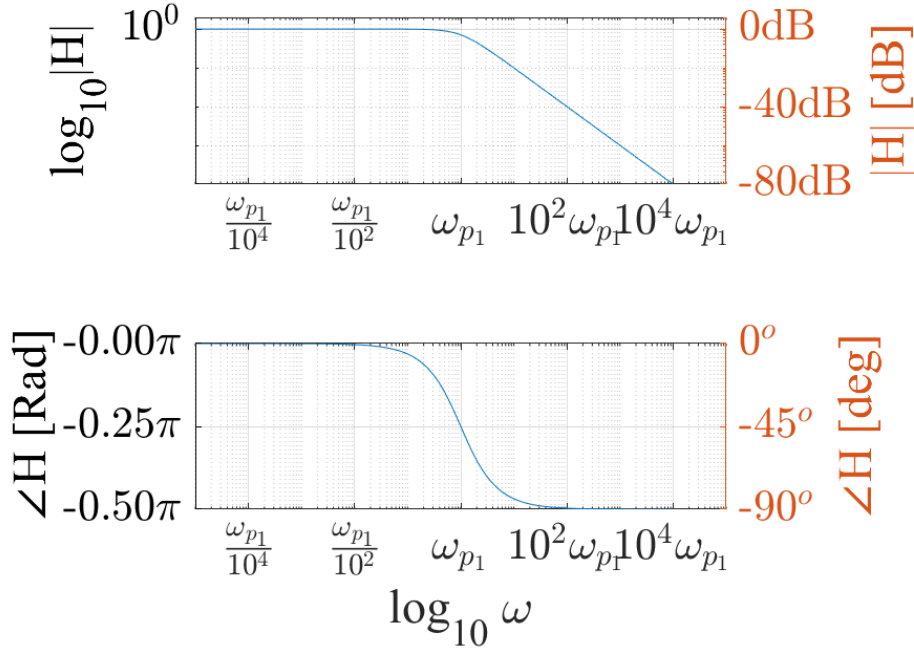


Figure 5.8: Contribution of a single pole to the Bode plot. Compare equation (5.19).

And if there is a real positive zero, let's say z_1 , then the corresponding term using the zero frequency would have a minus sign instead of a plus sign, i.e. $\dots(1 - \frac{s}{\omega_{z_1}})\dots$

Note that if one solution for $\frac{A}{a_o} = 0$ (or correspondingly $\frac{B}{b_o} = 0$) is $s = 0$, then there is one first order term in the product that is simply s , and a corresponding pole/zero frequency that is at zero. Also for convenience, let us choose to order these solutions by their magnitude, i.e. $\omega_{z_1} \leq \omega_{z_2} \leq \dots \leq \omega_{z_n}$ and $\omega_{p_1} \leq \omega_{p_2} \leq \dots \leq \omega_{p_m}$.

Let's look at the first pole and for now assume it is a negative real number. When evaluating the function H for $s = j\omega$, the term $1 + j\frac{\omega}{-\omega_{p_1}}$ does never become zero. Something else of interest is happening instead and the point when $\omega = \omega_{p_1}$ separates two distinct behaviours of the function $H(\omega)$ (compare Fig. 5.8) : as long as $|s| = \omega \ll \omega_{p_1}$ the term $\frac{1}{(1 + \frac{s}{\omega_{p_1}})} \approx 1$, so it does not affect the transfer function much. But as $|s| \gg \omega_{p_1}$ the term becomes $\frac{1}{(1 + \frac{s}{\omega_{p_1}})} \approx \frac{\omega_{p_1}}{s} = \frac{\omega_{p_1}}{j\omega} = -j\frac{\omega_{p_1}}{\omega}$. So this term adds hyperbolic decay to the magnitude $|H|$ (i.e. diminishing the magnitude of H proportionally with $\frac{1}{\omega}$) and also adds a $-\frac{\pi}{2}\text{Rad} = -90^\circ$ phase shift to H . When $\omega = \omega_{p_1}$ exactly, so between those two effects, the term is $\frac{1}{(1 + \frac{s}{\omega_{p_1}})} = \frac{1}{(1 + j)} = \frac{1}{\sqrt{2}e^{j\frac{\pi}{4}}} = \frac{1}{\sqrt{2}}e^{-j\frac{\pi}{4}}$, i.e. diminishing the magnitude by $\frac{1}{\sqrt{2}}$ and adding a phase shift of $-\frac{\pi}{4}\text{Rad} = -45^\circ$. So in summary for each real negative pole p_i it's corresponding contribution to the transfer function is:

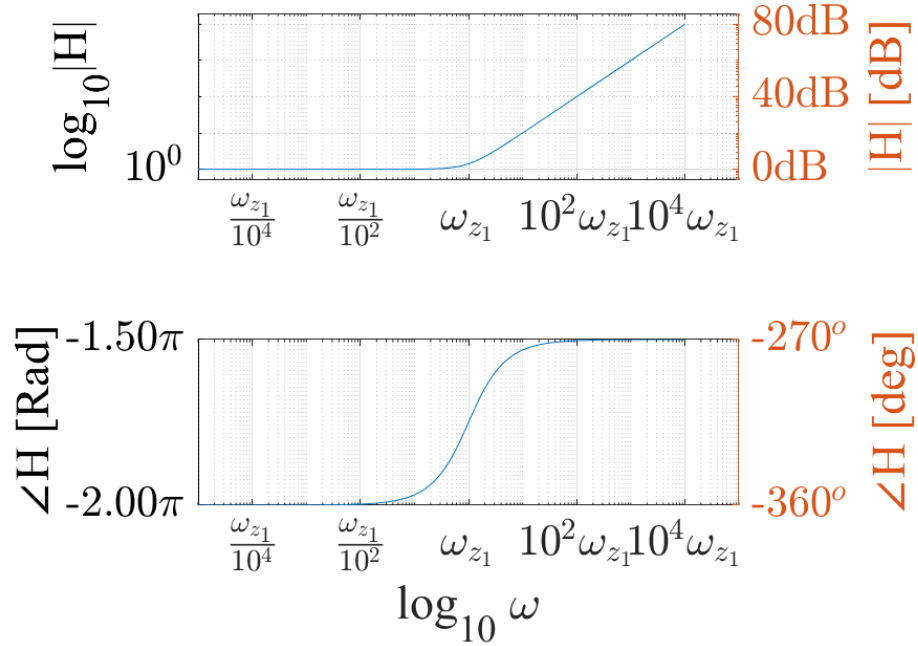


Figure 5.9: Contribution of a single *negative* zero to the Bode plot. Compare equation (5.21).

$$\frac{1}{1 + \frac{s}{\omega_{p_i}}} \begin{cases} \approx 1 & \text{if } |s| \ll \omega_{p_i} \\ = \frac{1}{\sqrt{2}} e^{-j\frac{\pi}{4}} & \text{if } |s| = \omega_{p_i} \\ \approx -j \frac{\omega_{p_i}}{\omega} & \text{if } |s| \gg \omega_{p_i} \end{cases} \quad (5.19)$$

And the angle it is contributing is:

$$\angle \left(\frac{1}{1 + \frac{s}{\omega_{p_i}}} \right) = -\arctan \frac{\omega}{\omega_{p_i}} \quad (5.20)$$

So a Bode plot of a single pole transfer function with no gain looks like figure 5.8. Individual contributions can be 'added' on the loglog magnitude plot, i.e. in fact they are multiplied. And also the contributions to the phase shift can be added, in this instance it's an addition for real.

Doing the same consideration for real negative zeros one gets (also see figure 5.9):

$$1 + \frac{s}{\omega_{z_i}} \begin{cases} \approx 1 & \text{if } |s| \ll \omega_{z_i} \\ = \sqrt{2} e^{j\frac{\pi}{4}} & \text{if } |s| = \omega_{z_i} \\ \approx j \frac{\omega}{\omega_{z_i}} & \text{if } |s| \gg \omega_{z_i} \end{cases} \quad (5.21)$$

and

$$\angle \left(1 + \frac{s}{\omega_{z_i}} \right) = \arctan \frac{\omega}{\omega_{z_i}} \quad (5.22)$$

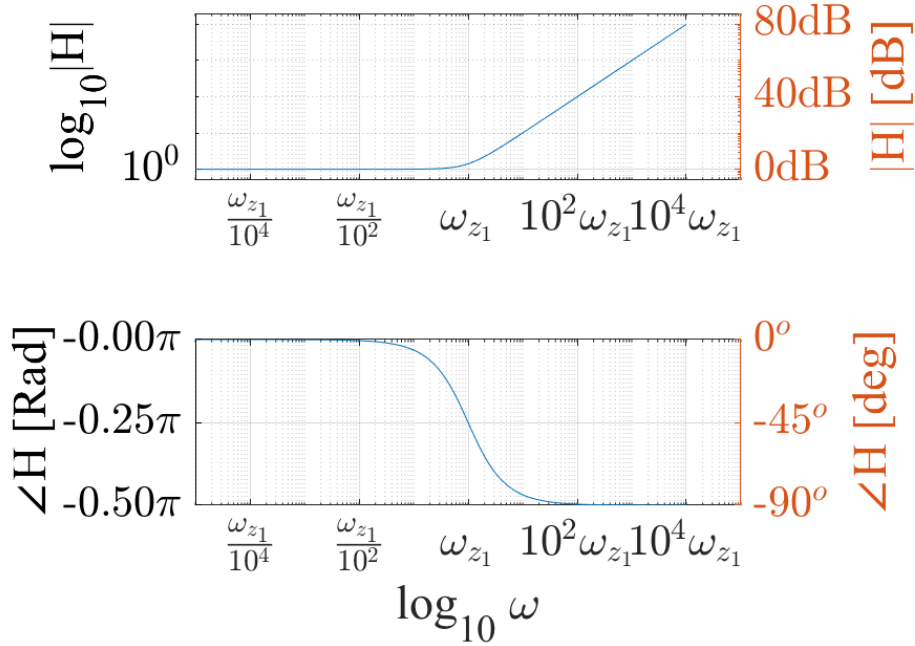


Figure 5.10: Contribution of a single *positive* zero to the Bode plot. Compare equation (5.23).

So the zeros (if they are real and negative) contribute a positive $\frac{\pi}{2}$ phase shift for frequencies higher than the pole frequency and hyperbolic growth proportional to ω .

Interestingly, *positive* real zeros do also commonly occur. In that case they also contribute hyperbolic growth, but now *negative* phase shift (like the negative real poles, also compare figure 5.10):

$$1 - \frac{s}{\omega_{z_i}} \begin{cases} \approx 1 & \text{if } |s| \ll \omega_{z_i} \\ = \sqrt{2}e^{-j\frac{\pi}{4}} & \text{if } |s| = \omega_{z_i} \\ \approx -j\frac{\omega}{\omega_{z_i}} & \text{if } |s| \gg \omega_{z_i} \end{cases} \quad (5.23)$$

and

$$\angle \left(1 - \frac{s}{\omega_{z_i}} \right) = -\arctan \frac{\omega}{\omega_{z_i}} \quad (5.24)$$

5.4.2 How to draw Bode plots from the real poles and zeros of a transfer function

(The 'real' in the section title is ment to distinguish from 'complex' poles that will be mentioned later in section 5.7.1.)

So lets go back to the Bode plot example in figure 5.7 and the corresponding transfer function already given in root form in (5.15). The equation has only two non-zero poles, and no zeros. So that makes the curve flat to start with.

At the first pole frequency the decay kicks in. Note that at the pole frequency the $\frac{1}{\sqrt{2}}$ reduction corresponds to the often used **-3dB** to refer to the **cut off frequency** (here we shall refer to it as ω_H) of a filter, i.e. the frequency at which the decay or **roll off** sets in and the power (proportional to the square of the amplitude) has been reduced to $\frac{1}{2}$. Note that the slope settles onto a straight line in the log-log plot of a 1:1 slope (i.e. x-times bigger frequency, means x-times smaller amplitude), or in dB the slope is 20dB per decade. Also on the phase plot it's the middle of a $-\frac{\pi}{2}$ Rad smooth step transition. At the next pole, the slope becomes steeper, since now $|H|$ diminishes with $\frac{1}{\omega^2}$. So for every multiplication of the frequency with x the magnitude shrinks by x^2 , which again is reflected in the log-log plot with a straight line but now falling 2:1 or 40dB per decade.

From those observations we can deduce a general set of rules how to draw Bode plots knowing the poles and zeros of a transfer function, provided all poles and zeros are real numbers. It is simply the superposition from all contributions of the individual poles and zeros:

1. In the magnitude plot there is a convex knee at each real pole frequency changing the slope by -20dB/decade, and concave knee for each real zero changing the slope by +20dB/decade.
2. In the phase plot there is a step down of $-\frac{\pi}{2}$ for each negative real pole and each positive real zero and a step up by $\frac{\pi}{2}$ for each negative real zero.
3. The curve is flat if it has an equal number of poles and zeros at lower frequencies (so for example with no zero or pole at point 0 the curve starts flat with the magnitude being $\frac{a_o}{b_o}$.)

5.5 Steps for deriving the transfer function and its Bode plot of a linear circuit

So in summary analysing a linear circuit will usually follow these steps:

1. Define input signal x and output signal y in the circuit!
2. Derive the Kirchoff current equations for each node of your circuit! (The voltage law can also be used or a combination.)
3. Solve the set of equations to get $h = \frac{y}{x}$!
4. Bring h into root form!
5. Draw the Bode plot using the poles and zeros!

5.6 Dominant pole approximation

For low pass filters, oftentimes one is mostly interested in the pole with the lowest frequency. If that first pole p_1 occurs at a frequency much lower (for practical purposes a factor 4 lower) than any other poles or zeros, the shape of the cut off 'knee' and thus the -3dB frequency and the filter's band width (BW) is entirely determined by that **dominant pole** alone.

In mathematical terms, one can find a dominant pole directly from the polynomial, if the first order parameter is much bigger than any higher order parameters, provided all pole frequencies are bigger than 1Rad. So if the transfer function's denominator is $b_0 + b_1s + b_2s^2 \dots b_ms^m$, then if there is a dominant pole that pole frequency is approximated by

$$\forall \{i > 1 : |b_i| \ll |b_1|\} \Leftrightarrow \frac{1}{\omega_{p_1}} \approx \frac{b_1}{b_0} \quad (5.25)$$

Just to illustrate *why* that is the case, we can look at a second order example, where the polynomial can be expressed as:

$$1 + \left(\frac{1}{\omega_{p_1}} + \frac{1}{\omega_{p_2}} \right) s + \frac{1}{\omega_{p_1}\omega_{p_2}} s^2 \quad (5.26)$$

So if $\omega_{p_2} \gg \omega_{p_1} > 1$ then the term $\frac{1}{\omega_{p_1}} + \frac{1}{\omega_{p_2}}$ is dominated by $\approx \frac{1}{\omega_{p_1}}$ and $\frac{1}{\omega_{p_1}\omega_{p_2}} \ll \left(\frac{1}{\omega_{p_1}} + \frac{1}{\omega_{p_2}} \right)$.

If one makes a stronger assumption about the dominant pole, namely that its frequency is smaller than any other poles' or zeros' by a factor corresponding to the DC gain $\forall \{i > 1, j : \omega_{p_i} * A_{DC} < \omega_{p_i} \wedge \omega_{p_i} * A_{DC} < \omega_{z_j}\}$, then any pole or zero occur only at frequencies beyond the **unity gain frequency** ω_T (i.e. where $|A(j\omega_T)| := 1 = A_{DC} * \frac{\omega_{p_1}}{\omega_T}$, because of the first order -20dB roll off until ω_T) and for all practical purposes one can simply treat the filter as a first order filter having a single pole only.

Hunting for a dominant pole in circuit schematics terms one can make some assumptions that eliminate the temporal dynamics from all but a single node (such as 'if this resistor is zero' or 'this capacitor is negligibly small relative to others'), assuming that the dominant pole occurs at that node. Check sections 9.4.3 and 9.4.4 for an example where we are looking for dominant poles in the general transconductance amplifier model under the assumption that the temporal dynamics in either the input or output node do dominate.

5.7 Stability

Linear filters where the signal is only processed in a feed forward fashion in all stages will not have stability issues. However, real electronic filter implementations will often include circular signal dependencies, i.e. contain some feedback loop. Such a feedback loop together with some amplification can potentially lead to self sustaining oscillations or even a 'run away' oscillation with infinitely increasing amplitude.

Unfortunately, even two of the one-transistor amplifying stages (see chapter 9.2) do have an element of feedback, since there is a capacitor between output and input for the CS and CD (i.e. SF) stages. Thankfully, the problem for the CS stage is mostly academic, but somewhat more real for the CD stage. We'll have a look at it as an example later on (subsection 9.6).

Furthermore, explicit negative feedback is an often used method in analog circuit design to obtain a precise gain. The designer is well advised to pay close attention to stability issues when employing this method. One would at first glance think that negative feedback per se is actually unproblematic, right? Unfortunately, negative feedback paired with delay/phase shift can in effect turn

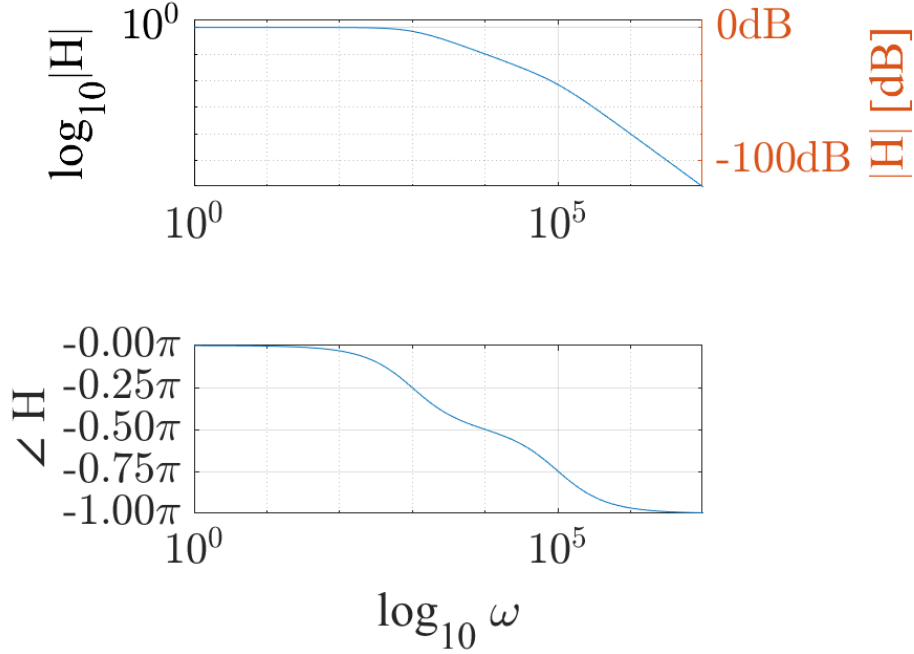


Figure 5.11: 2nd order low pass with $Q = 0.1$ and $\omega_0 = 10^4 \text{Rad}$.

into positive feedback under the right (i.e. wrong) conditions. So we'll look at some analysis tools to pinpoint the problem in the following.

5.7.1 2nd order equation with complex poles

So far we have discussed transfer functions with real poles and zeros. Another example Bode plot thereof with a second order polynomial denominator, no zeros and DC gain $A_{DC} = 1$ is shown in figure 5.11. However, it can occur that the polynomial in the denominator does have complex solutions and then the behaviour becomes more complicated and can indicate stability problems. We can tackle this for 2nd order polynomials in the denominator, i.e. transfer functions that give rise to complex poles. For this the 2nd order polynomial is by convention written as a function of two parameters, the **quality factor** Q and the **characteristic frequency** ω_0 :

$$H(s) = \dots \frac{\dots}{1 + \frac{1}{\omega_0 Q} s + \frac{1}{\omega_0^2} s^2} \quad (5.27)$$

So to find the poles one has to solve:

$$0 = 1 + \frac{1}{\omega_0 Q} s + \frac{1}{\omega_0^2} s^2 \quad (5.28)$$

Using the quadratic formula to find the poles p_1, p_2 , i.e solutions for s :

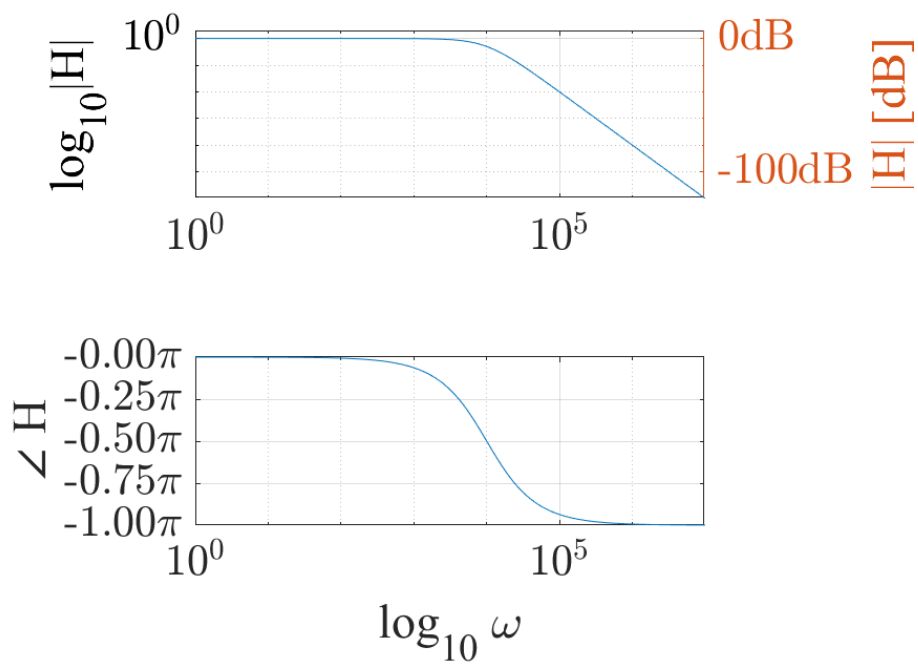


Figure 5.12: 2nd order low pass with $Q = 0.5$ and $\omega_0 = 10^4 \text{Rad}$. Highest Q resulting in two real poles, i.e. actually the very same pole twice.

$$p_1, p_2 = \frac{-\frac{1}{\omega_0 Q} \pm \sqrt{\frac{1}{\omega_0^2 Q^2} - 4\frac{1}{\omega_0^2}}}{2\frac{1}{\omega_0^2}} = \frac{-\frac{\omega_0}{Q} \pm \sqrt{\frac{\omega_0^2}{Q^2} - 4\omega_0^2}}{2} \quad (5.29)$$

Looking at the expression under the square root, it becomes negative when

$$\frac{1}{Q^2} < 4 \Leftrightarrow |Q| > 0.5 \quad (5.30)$$

In that case p_1 and p_2 will be a conjugate complex pair, and their magnitude will be exactly $|p_{1,2}| = \omega_0$. So figure 5.11 shows an example Bode plot for a smaller $Q = 0.1$ and $\omega_0 = 10^4$, with two real poles. The poles are centred around ω_0 . Then figure 5.12 shows a Bode plot example where $Q = 0.5$ and $\omega_0 = 10^4$. Now there are still two real poles at the exact same frequency $\omega_{p_1} = \omega_{p_2} = \omega_0$. The Bode plot still looks like we are used to, following the instructions for drawing Bode plots with real poles. Since there are two poles at $\omega = 10^4$, we get a drop of -6dB at that point and the roll off will immediately be $\frac{40\text{dB}}{\text{dec}}$.

Now if Q increases some more we get a *conjugate complex* (!) pair of poles. There starts to be a subtle deviation from Bode plots with just real poles, hardly noticeable at first. What happens is that the gain at the cut off frequency increases, i.e. that the -6dB point, which is at ω_0 for two identical real poles, i.e. $Q = 0.5$, is shifted to slightly higher frequencies, while the roll off for very large frequencies remains in place. Another way of saying this is that the 'knee' becomes sharper. This continues until the gain is at -3dB at ω_0 , i.e. like for a 1st order LP, but with a second order roll off of $\frac{40\text{dB}}{\text{dec}}$. That point is reached when $\angle -p_{1,2} = \pm 45^\circ$. So the angles of the conjugate complex pair of the inverse of the poles (as they occur in the root form) are $\angle -p_1 = -\angle \frac{1}{-p_1} = -45^\circ$ and $\angle -p_2 = -\angle \frac{1}{-p_2} = 45^\circ$ (see figure 5.13), the imaginary $\pm \sqrt{\frac{\omega_0^2}{Q^2} - 4\omega_0^2}$ part of the poles is the same magnitude as the real part $-\frac{\omega_0}{Q}$. The condition for that is:

$$\left| -\frac{\omega_0}{Q} \right| = \left| \sqrt{\frac{\omega_0^2}{Q^2} - 4\omega_0^2} \right| \Leftrightarrow Q = \frac{1}{\sqrt{2}} \Leftrightarrow \angle \frac{1}{-p_{2,1}} = \pm 45^\circ \quad (5.31)$$

This is the point of **maximally flat response**, i.e. the BW gets extended by the slight resonance around and beyond the cut off frequency, but the gain still does not exceed the DC gain. See figure 5.14!

To prove this let's look at how the denominator $(1 + \frac{j\omega}{-p_1})(1 + \frac{j\omega}{-p_2})$ of the transfer function behaves for increasing ω for when $Q = \frac{1}{\sqrt{2}}$ and then for when $Q > \frac{1}{\sqrt{2}}$. Let us define a variable name for the two multipliers of the denominator $A := 1 + \frac{j\omega}{-p_1}$ and $B := 1 + \frac{j\omega}{-p_2}$, as well as a variable c that is the magnitude of the real and imaginary part (they are the same length when $Q = \frac{1}{\sqrt{2}}$) of the expression $\frac{j\omega}{-p_{1,2}}$, i.e. $c := |\text{Re}(\frac{j\omega}{-p_{1,2}})| = |\text{Im}(\frac{j\omega}{-p_{1,2}})|$. Then it can be shown that:

$$\begin{aligned} |A * B|^2 &= ((1+c)^2 + c^2) * ((1-c)^2 + c^2) \\ &= 1 + 4c^4 \end{aligned} \quad (5.32)$$

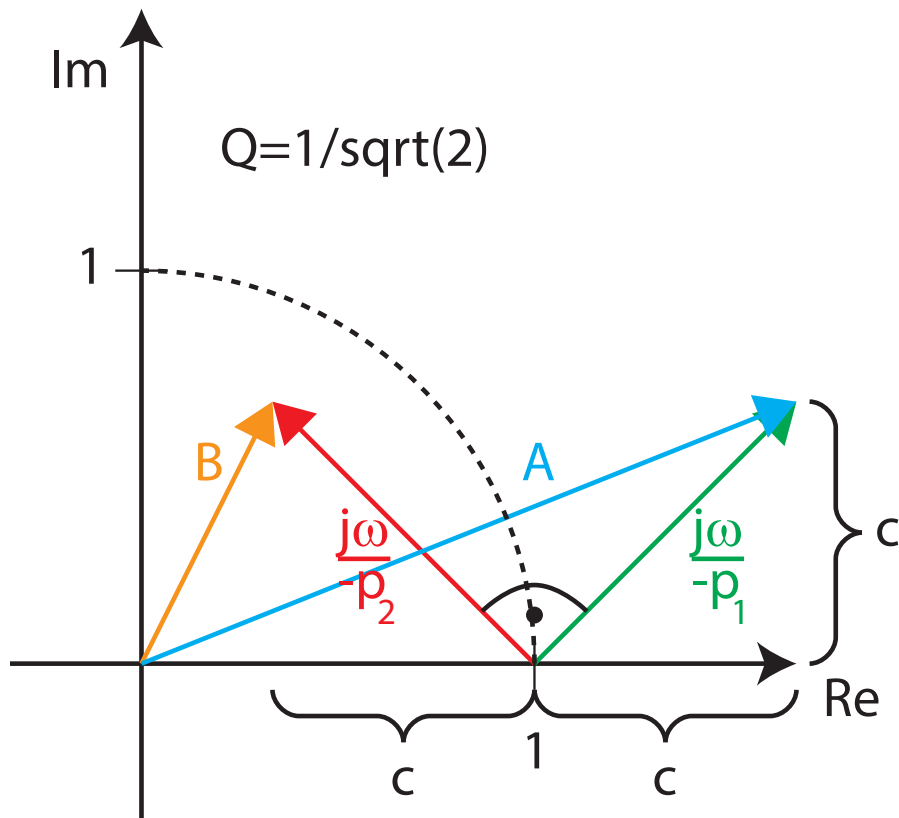


Figure 5.13: Illustration of what happens to the denominator $A * B$ in the complex plain of a 2nd order low pass filter transfer function with $Q = \frac{1}{\sqrt{2}}$ for increasing $j\omega$. Maximally flat response.

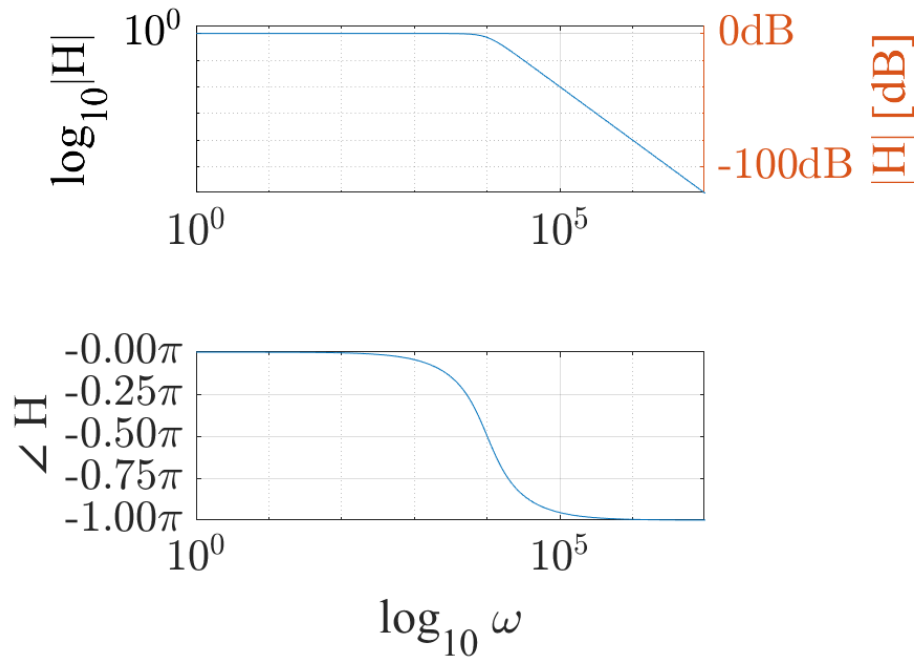


Figure 5.14: 2nd order low pass with $Q = \frac{1}{\sqrt{2}} = 0.71$ and $\omega_0 = 10^4 \text{Rad}$. Two complex poles, a conjugate complex pair. Maximally flat response, i.e. no resonance just yet.

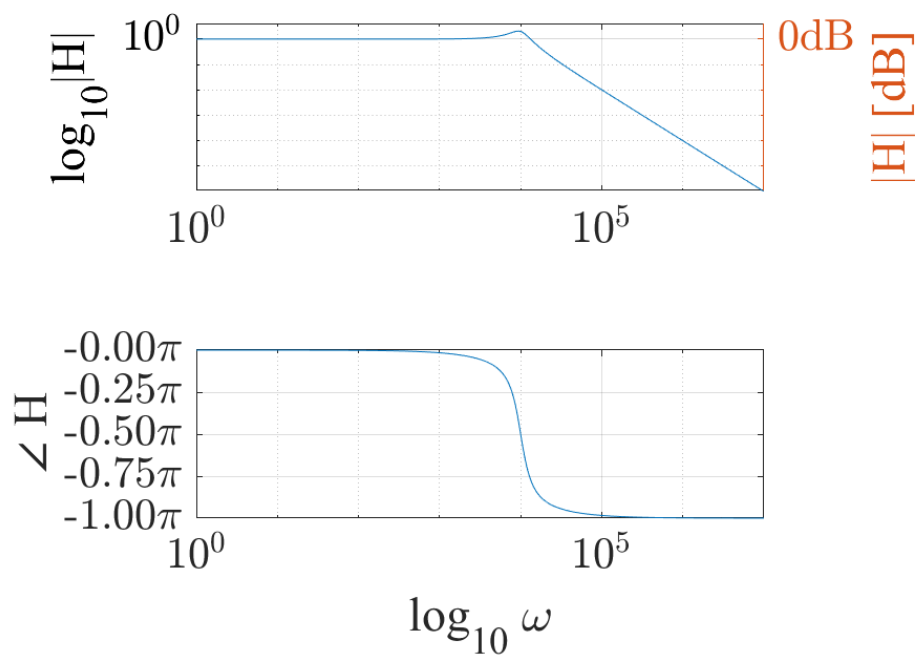


Figure 5.15: 2nd order low pass with $Q = 2$ and $\omega_0 = 10^4\text{Rad}$. Substantial resonance peak (not so prominent on the logarithmic scale: the peak looks much more pronounced on a linear scale)

So the magnitude of the denominator of the transfer function grows proportionally with $\sqrt{1 + 4c^4}$ if $Q = \frac{1}{\sqrt{2}}$. In other words it is never smaller than 1, and thus its inverse is never bigger than 1, so no resonance and no 'hump' in the transfer function.

But if $Q > \frac{1}{\sqrt{2}}$ is just a little bigger, then we have $|\text{Im}(-p_1)| > |\text{Re}(-p_1)|$ or $|\text{Im}(\frac{1}{-p_1})| > |\text{Re}(\frac{1}{-p_1})|$ or $\text{Im}(\frac{j\omega}{p_{1,2}}) < \left| \text{Re}(\frac{j\omega}{-p_{1,2}}) \right|$. We define thus the ratio:

$$r := \frac{\left| \text{Re}(\frac{j\omega}{-p_{1,2}}) \right|}{\text{Im}(\frac{j\omega}{-p_{1,2}})} \quad (5.33)$$

And

$$Q > \frac{1}{\sqrt{2}} \Leftrightarrow r > 1 \quad (5.34)$$

It can be shown that:

$$\frac{1}{|A * B|} > 1 \Leftrightarrow \frac{2(r^2 - 1)}{(r^2 + 1)^2} > \text{Im}(\frac{j\omega}{-p_{1,2}})^2 \quad (5.35)$$

so only for $r > 1$ this equation has a solution (because $\text{Im}(\frac{j\omega}{-p_{1,2}})^2$ is always bigger than zero and $\frac{2(r^2 - 1)}{(r^2 + 1)^2}$ is bigger than zero if $r > 1$) and there exists an upper limit for ω where $\frac{1}{|A * B|} > 1$ and we do have some resonance at frequencies up to that limit¹, which becomes visibly pronounced in the Bode plot close to the cut off frequency shortly before the roll off starts as in the example Bode plot in figure 5.15. Interestingly the band where you do get $\frac{1}{|A * B|} > 1$ is widest for $r = \sqrt{3}$.

So now we have analysed a second order low pass transfer function and derived a condition for it having some resonance, most pronounced around the cut off region. Note that 2nd order low pass functions, even if they do express some resonance, will still not be in danger of a 'run away' amplification: the resonance will always be limited, i.e. $\frac{1}{|A * B|} < \infty$.

But what to do if the circuit analysis yields a higher order denominator? Well, there is a bunch of methods to approximate a higher order polynomial with a lower order polynomial. We have mentioned the dominant pole approximation already (See page 62), which approximates a polynomial with a first order polynomial, if the higher order parameters of the polynomial are small (if the pole and zero frequencies are bigger than 1Hz!). You can do the same, if the polynomial's parameters of order higher than two are much smaller than the first two, and reduce the polynomial to a second order polynomial, ignoring the higher orders.

There is also a mathematical somewhat more precise way of approximating a higher order transfer function in root form with a 2nd order function where you use an equivalent pole frequency ω_{eq} as the 2nd pole frequency. This method however, requires that you *know* what all the higher order poles are and is thus not helpful for stability analysis and/or finding an approximate Q-factor. It is described in section 5.8 equation (5.45). It IS helpful to find good approximation

¹and if $r < 1$ the condition cannot be fulfilled, ergo no resonance up to that point of r .

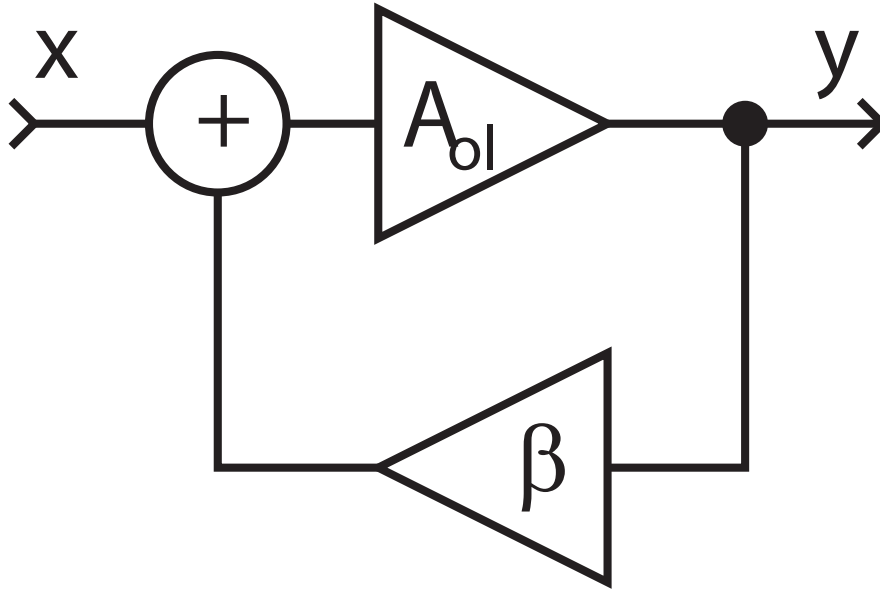


Figure 5.16: General concept of a feedback loop. $A_{ol}(x(s))$ is the **open loop gain** and frequency dependent transfer function. β is often considered to be a frequency independent constant, but can be frequency dependent as well. The resulting **closed loop gain** A_{cl} is stabilising ($A_{cl} < \infty$) if the total loop gain $A_{ol}\beta < 1$. It is smaller than the open loop gain $A_{cl} < A_{ol}$ if the loop gain is negative $A_{ol}\beta < 0$, and is approaching $A_{cl} \approx \frac{1}{\beta}$ if the loop gain is $A_{ol}\beta \ll -1$. NOTE: due to the frequency dependency of A_{ol} the feedback gain A_{cl} can change its behaviour too in dependency of frequency, i.e. become unstable only for certain frequencies!

of the -3dB cut off frequency when the higher order poles and zeros are close to the second order pole.

So instead, we shall in the following look at a more applied method to gauge stability, where it is sufficient to have the transfer characteristics (i.e. the Bode plots) of some circuit building blocks derived from simulation, i.e. it is not necessary to have an explicit root form representation of the transfer function.

5.7.2 Negative feedback, loop gain, and phase margin

A different approach to analyse stability issues of electronic filters due to a feedback loop is possible even for higher order filters without dominant poles if one can pinpoint the feedback loop and disconnect it. This becomes trivial if the circuit is designed that way to begin with. That is, one starts with a filter with a known transfer function (i.e. with a so called (usually frequency dependent) **open loop gain** A_{ol} and connects it in an explicit feedback loop. In fact this is a popular method of designing amplifiers with a specific **closed loop gain** $A_{cl} := \frac{y}{x}$, using a generic operational amplifier with a very high gain in a (negative) feedback loop as illustrated in figure 5.16. Such a feedback

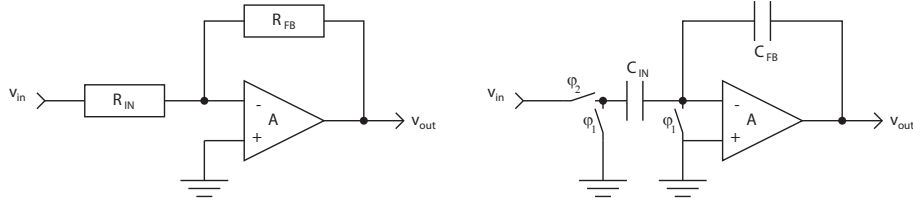


Figure 5.17: Two examples of amplifier circuits employing negative feedback loops to obtain a specific gain. To the left with resistive feedback and to the right with capacitive feedback in a switched capacitor setup. The switched capacitor circuit is 'reset' when switches φ_1 are conducting. Then they are disconnected and switch φ_2 is connected, moving the left hand terminal of C_{IN} from signal ground to v_{in} and thus v_{out} will move from signal ground (actually from the offset voltage of the amplifier for zero input in a real implementation) by $A_{cl}v_{in}$ volts. To draw the analogy to figure 5.16, the input x is respectively $x = v_{in} \frac{R_{FB}}{R_{FB}+R_{IN}}$ on the left and $x = v_{in} \frac{C_{IN}}{C_{IN}+C_{FB}}$ on the right. β is respectively $\beta = \frac{R_{IN}}{R_{FB}+R_{IN}}$ on the left and $\beta = \frac{C_{FB}}{C_{IN}+C_{FB}}$ on the right. A_{ol} is $A_{ol} = -A$. If $A = \infty$ and assuming no frequency dependencies the closed loop gain on the left with respect to x is $-\frac{1}{\beta} = -\frac{R_{FB}+R_{IN}}{R_{IN}}$, but with respect to v_{in} it is $A_{cl} = -\frac{R_{FB}+R_{IN}}{R_{IN}} \frac{R_{FB}}{R_{FB}+R_{IN}} = -\frac{R_{FB}}{R_{IN}}$. By the same considerations the closed loop gain of the switched capacitor circuit with respect to V_{IN} is $A_{cl} = \frac{C_{IN}+C_{FB}}{C_{FB}} \frac{C_{IN}}{C_{IN}+C_{FB}} = \frac{C_{IN}}{C_{FB}}$.

loop is stabilizing for any specific frequency if the **loop gain** $A_{ol}\beta < 1$ at that frequency. In particular it will stabilize at an absolute gain smaller than $|A_{ol}|$ if the feedback loop is negative, i.e. $A_{ol}\beta < 0$, and will stabilize with resonance at a level higher than $|A_{ol}|$ for $0 < A_{ol}\beta < 1$. But it will NOT stabilize and end up oscillating uncontrolled if $A_{ol}\beta \geq 1$.² More precisely:

$$A_{cl} := \frac{y}{x} = \begin{cases} \infty \text{ (unstable)} & \text{if } A_{ol}\beta \geq 1 \\ \frac{A_{ol}}{1-A_{ol}\beta} & \text{if } A_{ol}\beta < 1 \end{cases} \quad (5.36)$$

and notably for the clearly stable cases, i.e. where the loop gain is very negative ($A_{ol}\beta \ll -1$):

$$A_{cl} \approx -\frac{1}{\beta} \quad \text{if } A_{ol}\beta \ll -1 \quad (5.37)$$

This is actually a method of deriving specific gains from a high gain operational amplifier: in practical terms it is often difficult to design a CMOS amplifier with a specific open loop gain. So rather design one with as high a gain as possible and use resistive division or capacitive division in a switch cap design to construct a specific β to get a rather precise closed loop gain $A_{cl} \approx -\frac{1}{\beta}$, where the approximation becomes more precise, the higher the open loop gain

²For $A_{ol}\beta > 1$, $A_{cl} = \frac{A}{1-A\beta}$ is as a matter of fact also a fixed point, but it is a 'repulsive' fixed point, i.e. the circuit actually receives recursive positive feedback and will be forced away from that fixed point at even the smallest deviation and become unstable. So from a random starting point it would not be able to stabilize at all and instead be driven to ∞ ...

is. See figure 5.17 for the typical examples. The deviation of A_{cl} from $\frac{1}{\beta}$ is often referred to as **closed loop gain error**:

$$A_{cl} - \left(-\frac{1}{\beta}\right) = \frac{A_{ol}}{1 - A_{ol}\beta} + \frac{1}{\beta} = \frac{\beta A_{ol} + 1 - A_{ol}\beta}{\beta(1 - A_{ol}\beta)} = \frac{1}{\beta} \frac{1}{1 - A_{ol}\beta} \quad (5.38)$$

So far so good, but what happens when frequency dependencies enter the picture? Let's just consider the case where there is only frequency dependency in $A_{ol}(\omega)$ but no frequency dependency in $\beta(\omega) = \beta$. Even if one designs such a system to be stable at DC and low frequencies with a negative loop gain well within the 'safe zone' $A_{ol}(0\text{Rad})\beta < 0$, well, the situation becomes complicated when starting to include complex number arithmetic into the equation. The kind of circuits we look at here are low pass filters. For those, increase in frequency brings a reduction in gain and at the same time a phase shift. Let's assume we have a save stable behaviour at DC with $A_{ol}(0\text{Rad})\beta < 0$, i.e. the phase shift of the loop gain to begin with is $\angle A_{ol}(0\text{Rad})\beta = -180^\circ$. At higher frequencies we may get additional phase shift and possibly at a particular frequency ω_{critical} , that phase shift might accumulate to an additional $-\pi\text{Rad} = -180^\circ$. If this happens the loop gain at that frequency is now once more real and *positive* $A_{ol}(\omega_{\text{critical}})\beta > 0$ (i.e. $\angle A_{ol}(\omega_{\text{critical}})\beta = -360^\circ$). As we know from (5.36), if the loop gain is positive the loop can still be stable if, and only if, it is smaller than 1! Thankfully, phase shift in a low pass filter comes together with a roll off in gain. So if we are lucky the roll off ensures that $A_{ol}(\omega_{\text{critical}})\beta < 1$, but there is in fact a 'race' between roll off and phase shift.

So in negative feedback closed loops (!) we can be sure to be safe when

1. A_{ol} is 2^{nd} or first order. In that case the added phase shift will always remain smaller than $-\pi = -180^\circ$, i.e. the total phase shift will never quite reach the critical value $-2\pi = -360^\circ$. Note that there may still be *resonance* for 2nd order transfer functions (see the section 5.7.1 on the Q-factor) but that resonance will be limited.
2. if we make sure that the decrease in gain is sufficient and $|A_{ol}(\omega_{\text{critical}})\beta| < 1$. Also here there may still be limited resonance or even instability (i.e. unlimited or 'run away' resonance) if you have very high gain still close to ω_{critical} : This only avoids a guaranteed instability e.g the rule is required but not sufficient)! So usually one is well advised to aim at $|A_{ol}(\omega_{\text{critical}})\beta| \ll 1$ by a good margin (see 'phase margin' (PM) later in this section).

The latter can be achieved by the choice of β : beta needs to be small enough to avoid this, i.e. the closed loop gain needs to be small enough at DC to give the roll off a head start over the phase shift. See figure 5.18 for an illustration: the phase shift at the loop gain unity gain frequency ω_T is where the loop gain $A_{ol}\beta = 1$ and consequently where $A_{ol} = \frac{1}{\beta}$ (as marked in figure 5.18) and thus the point where the additional phase shift should not yet have reached -180° (or the total phase shift should not have reached -360°) for the feedback system to avoid guaranteed instability. However, usually one is not at liberty to choose β freely: it is after all the parameter that sets the close loop gain $A_{cl} \approx \frac{1}{\beta}$.

Another method to keep the additional phase shift above -180° when the loop gain reaches $A_{ol}\beta = 1$ is **pole separation**, i.e. making sure A_{ol} has a

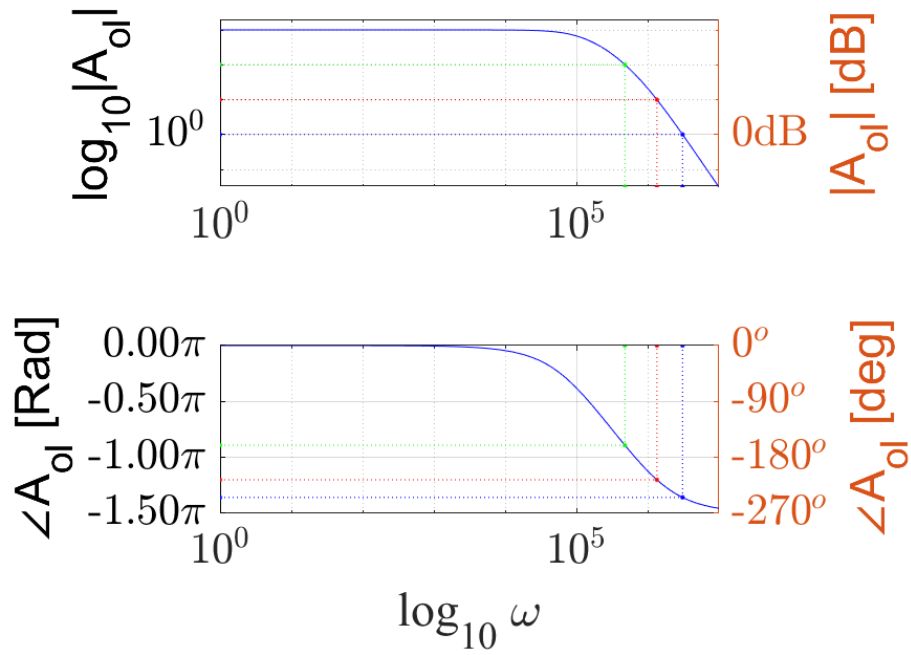


Figure 5.18: Illustrating varying the magnitude of $|\beta|$ to shift the unity gain frequency of the loop gain to lower frequency and thereby increase the **phase margin (PM)**. The transfer function used is a 3^{rd} order low pass filter without zeros, where $A_{ol} = 1000$, $\omega_{p1} = 10^5$, $\omega_{p2} = 10^{5.5}$, and $\omega_{p3} = 10^6$. The dotted lines mark the open loop gain for which the loop gain will be -1, the blue for $\beta = -1$, the red for $\beta = -\frac{1}{10}$ and the green for $\beta = -\frac{1}{100}$. The green results in additional phase shift of $-0.78\pi\text{Rad} = -139^\circ$ at the unity loop gain, i.e. as $A_{ol} = \frac{1}{\beta}$, leading to a positive phase margin of 41° . The other phase shifts at unity loop gain exceed -180° and do thus have negative PM and are guaranteed unstable.

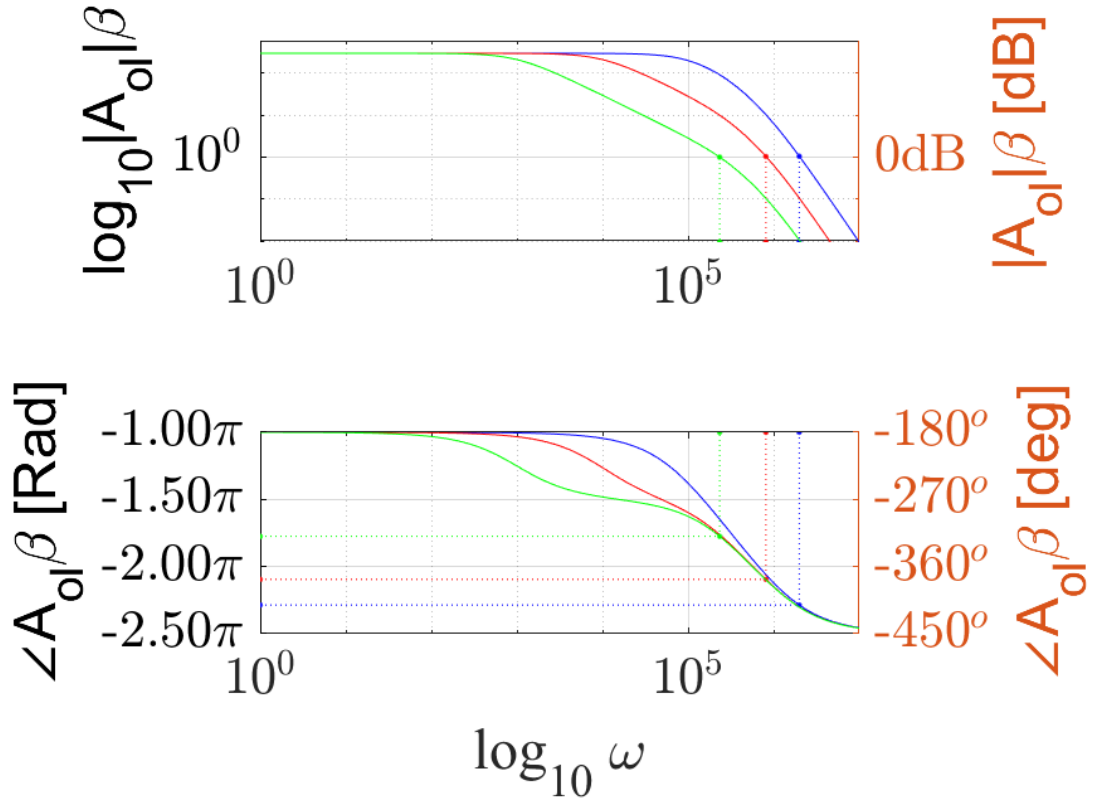


Figure 5.19: Illustrating pole separation to shift the unity gain frequency ω_T of the loop gain to lower frequencies increasing phase margin. The three bode plots are from a 3^{rd} order low pass loop (!) gain with no zeros and have all $A_{ol}(0)\beta = -300$, and $\omega_{p_2} = 10^{5.5}$ and $\omega_{p_3} = 10^6$. However, ω_{p_1} varies and is $\omega_{p_1} = 10^5$ for the blue, $\omega_{p_1} = 10^4$ for the red, and $\omega_{p_1} = 10^3$ for the green curve. So as the magnitude reaches 0dB, the phase shift is at $-2.29\pi\text{Rad} = -412^\circ$, $-2.10\pi\text{Rad} = -377^\circ$ and $-1.78\pi\text{Rad} = -319^\circ$. So only the green loop gain will lead to a stable feedback loop with a phase margin (PM) of $-319^\circ + 360^\circ = 41^\circ$

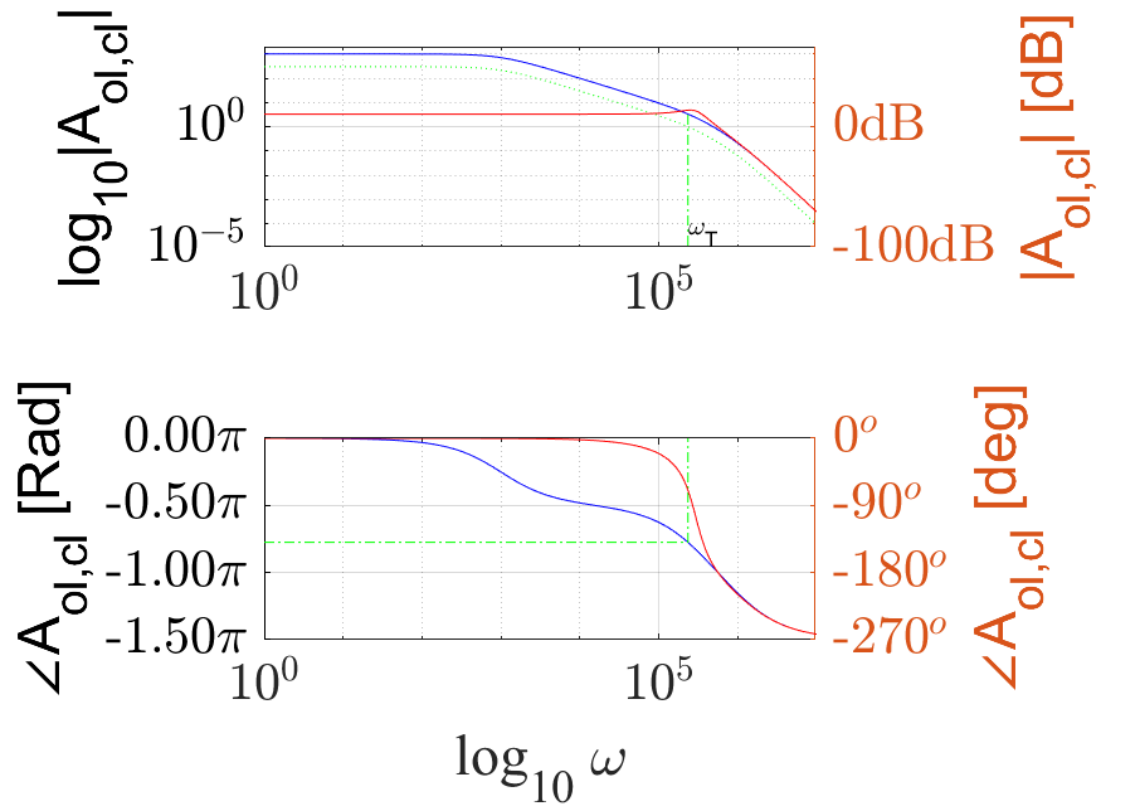


Figure 5.20: Illustrating the BW of a closed loop negative feedback system, for $A_{ol}(0) = 1000$, $\beta = -\frac{3}{10}$, $p_1 = 10^3$, $p_2 = 10^{5.5}$, and $p_3 = 10^6$. A_{ol} is in blue and A_{cl} in red. The dotted green line shows the loop gain and the dash-dotted line marks the unity gain of the loop gain and the additional phase shift at that frequency, which is $-2.43\text{Rad} = -139^\circ$. The phase margin here is thus 41° .

clearly dominant pole, for instance by adding capacitance to the node corresponding to the lowest pole. This way one gives again a head start to the roll off, but this time starting it at lower frequency instead of lowering its starting point. See figure 5.19 for an illustration of the effect on the loop gain if you lower the frequency of the first pole. If there is a clearly dominant pole, the roll off will start at that pole's frequency and the additional -180° phase shift will only occur somewhere beyond the second pole, at which point the roll off after the dominant pole will have had much 'time' to reduce the magnitude, hopefully below the loop gain's **unity gain frequency** ω_T , i.e. the frequency at which the magnitude of the loop gain $|A_{ol}(\omega_T)\beta| = 1 \Leftrightarrow |A_{ol}| = |\frac{1}{\beta}|$ has shrunk to unity. Also this method has a trade off: shifting the first pole of the loop gain to a lower frequency may also affect the closed loop bandwidth.

The **phase margin (PM)** is a measure of how little resonance the feedback loop system is expected to express. It is the difference between the phase shift of the loop gain at the unity gain frequency and the critical total $= -360^\circ = -2\pi$ Rad phase shift.

$$\text{PM} := \angle A_{ol}(\omega_T)\beta - \angle A_{ol}(\omega_{\text{critical}})\beta = \angle A_{ol}(\omega_T)\beta + 2\pi \quad (5.39)$$

The more positive the PM, the more likely the system will be stable, and if the PM of a second order LP filter (also for higher order systems if all other poles are way beyond ω_T) is bigger than 65° , it follows that $Q < \frac{1}{\sqrt{2}}$ (maximally flat response) and there will be no resonance. In fact there is a few more exact statements one can make, assuming that the A_{ol} is second order.

NOTE that the loop gain's $A_{ol}\beta$ bandwidth is the same as the open loop A_{ol} BW, but NOT THE SAME as the closed loop $A_{cl} = \frac{A_{ol}}{1-A_{ol}\beta}$ bandwidth: generally the cut off of A_{cl} is at the frequency where the open loop gain is reduced to $A_{ol} = \frac{1}{\beta}$. So the BW of A_{cl} will also be reduced when the BW of A_{ol} is reduced by pole separation but will still be larger. See figure 5.20 for an illustration. It shows A_{ol} in blue and A_{cl} in red, again for A_{ol} being a third order LP filter with no zeros where $A_{ol}(0) = 1000$, $\beta = -\frac{3}{10}$, $\omega_{p1} = 10^3$, $\omega_{p2} = 10^5.5$, and $\omega_{p3} = 10^6$. So it corresponds to the green graph in figure 5.19. With a PM of 41° it does experience some resonance. Both the magnitude and the phase plot approximate $\frac{1}{\beta}$ for very low frequencies and A_{ol} for very high frequencies, more precisely the separating frequency of the two is the frequency at which $A_{ol} = \frac{1}{\beta}$.

$$A_{cl}(\omega) \approx \begin{cases} \frac{1}{\beta} & \text{if } A_{ol} \gg \frac{1}{\beta} \\ A_{ol} & \text{if } A_{ol} \ll \frac{1}{\beta} \end{cases} \quad (5.40)$$

Assuming 2^{nd} order open and closed loop gain

Assuming that A_{ol} is a second order LP filter with no resonance (i.e. $Q_{ol} < 0.5$ and the open loop poles are real) one can deduce some properties of the closed loop system. For instance, one can deduce the poles and Q_{cl} and ω_{0cl} from the poles of A_{ol} .

Let's say that

$$\begin{aligned}
A_{ol}(s) &= A_{ol}(0) \frac{1}{\left(1 + \frac{s}{\omega_{p1}}\right)\left(1 + \frac{s}{\omega_{p2}}\right)} \\
&= A_{ol}(0) \frac{1}{1 + \left(\frac{1}{\omega_{p1}} + \frac{1}{\omega_{p2}}\right)s + \frac{1}{\omega_{p1}\omega_{p2}}s^2}
\end{aligned} \tag{5.41}$$

Then we solve for the A_{cl} poles. Let's define variables for nominator and denominator of the open loop gain $\frac{A_n}{A_d} := A_{ol}$. A bit of manipulation of equation (5.41) reveals that we only need to solve for the denominator to be zero. For stable cases we get from 5.36:

$$\begin{aligned}
A_{cl} &= \frac{\frac{A_n}{A_d}}{1 - \frac{A_n}{A_d}\beta} \\
&= \frac{A_n}{A_d \left(1 - \frac{A_n}{A_d}\beta\right)} \\
&= \frac{A_n}{A_d - A_n\beta}
\end{aligned} \tag{5.42}$$

solving for $0 = A_d - A_n\beta$ is the same as solving for $0 = 1 - \frac{A_n}{A_d}\beta$, consequently we solve:

$$0 = 1 + A_{ol}\beta \tag{5.43}$$

to get the closed loop poles

$$\begin{aligned}
0 &= 1 - A_{ol}(0) \frac{1}{1 + \left(\frac{1}{\omega_{p1}} + \frac{1}{\omega_{p2}}\right)s + \frac{1}{\omega_{p1}\omega_{p2}}s^2} \beta \\
0 &= 1 - A_{ol}(0)\beta + \left(\frac{1}{\omega_{p1}} + \frac{1}{\omega_{p2}}\right)s + \frac{1}{\omega_{p1}\omega_{p2}}s^2 \\
&\Rightarrow \\
p_{cl1}, p_{cl2} &= \frac{-(\omega_{p1} + \omega_{p2}) \pm \sqrt{(\omega_{p1} + \omega_{p2})^2 - 4\omega_{p1}\omega_{p2}(1 - A_{ol}(0)\beta)}}{2} \tag{5.44} \\
&\Rightarrow \\
Q_{cl} &= \frac{\sqrt{\omega_{p1}\omega_{p2}(1 - A_{ol}(0)\beta)}}{\omega_{p1} + \omega_{p2}} \\
\omega_{0cl} &= \sqrt{\omega_{p1}\omega_{p2}(1 - A_{ol}(0)\beta)}
\end{aligned}$$

One can also establish a relation between Q_{cl} , PM, $\frac{\omega_T}{\omega_{p_{ol2}}}$ as shown in table 5.3

5.8. APPROXIMATING HIGHER ORDER WITH 1st ORDER OR 2nd ORDER 79

PM	35.09	40.65	50.35	54.18	60.75	64.02	70.23	75.54	79.55	85.00
$\frac{\omega_T}{\omega_{p_{ol2}}}$	1.48	1.20	0.85	0.74	0.58	0.50	0.37	0.27	0.20	0.10
Q	1.60	1.35	1.05	0.97	0.82	0.75	0.63	0.53	0.45	0.32

Table 5.3: For a second order A_{ol} : Relation between the phase margin PM, the ratio of the loop gain A_{ol}/β unity gain frequency ω_T and the open loop 2nd pole frequency $\omega_{p_{ol2}}$ (For higher order A_{ol} one may employ the 2nd order approximation as derived with (5.45), but if the higher order poles are close to $\omega_{p_{ol2}}$ the PM in this table will deviate significantly), and the closed loop Q -factor.

5.8 Approximating higher order with 1st order or 2nd order

So if one is interested in these precise statements that are possible for 2nd order A_{ol} and A_{cl} and in general wants to define a Q for higher order functions, one can approximate higher order functions with 2nd order or even just 1st order. (The later makes sense in case of a very dominant pole.)

One can more generally approximate n^{th} order root form transfer function with an x^{th} ($x < n$) order root form transfer function by replacing the x^{th} pole frequency multiplier in the denominator ($1 + \frac{s}{\omega_{p_x}}$) and all higher frequency pole multipliers where $\omega_{p_i} \geq \omega_{p_x}$ and higher frequency zero multipliers $\omega_{z_i} \geq \omega_{z_y}$ in the nominator (where we denote ω_{z_y} as the smallest (!) of these zero frequencies with) with a single multiplier in the denominator ($1 + \frac{s}{\omega_{eq}}$) where ω_{eq} is computed as:

$$\omega_{eq} \approx \frac{1}{\sqrt{\sum_{\omega_{p_i} \geq \omega_{p_x}} \frac{1}{\omega_{p_i}^2} - \sum_{\omega_{z_i} > \omega_{p_x}} \frac{2}{\omega_{z_i}^2}}} \quad (5.45)$$

To write this out:

$$A \frac{(1 + \frac{s}{\omega_{z_1}}) \dots (1 + \frac{s}{\omega_{z_m}})}{(1 + \frac{s}{\omega_{p_1}}) \dots (1 + \frac{s}{\omega_{p_n}})} \approx A \frac{(1 + \frac{s}{\omega_{z_1}}) \dots (1 + \frac{s}{\omega_{z_{y-1}}})}{(1 + \frac{s}{\omega_{p_1}}) \dots (1 + \frac{s}{\omega_{p_{x-1}}}) (1 + \frac{s}{\omega_{eq}})} \quad (5.46)$$

This formula is obtained by assuming that the pole- and zero frequencies larger than ω_{p_x} are in general larger than 1, as well as only considering the poles and zeros at higher frequencies than ω_{p_x} and solving for the frequency where the magnitude of that expression H being $|H| = \frac{1}{\sqrt{2}}$. For example when approximating a second order expression with a first order, i.e. approximating the cut off frequency ω_H where the magnitude is reduced to $|H| = \frac{1}{\sqrt{2}}$ with the frequency ω_{eq} :

$$|H(\omega_H)|^2 = \frac{1}{2} = \frac{(1 + \frac{\omega_H^2}{\omega_{z1}^2})(1 + \frac{\omega_H^2}{\omega_{z2}^2})}{(1 + \frac{\omega_H^2}{\omega_{p1}^2})(1 + \frac{\omega_H^2}{\omega_{p2}^2})} \quad (5.47)$$

$$= \frac{1 + \omega_H^2 \left(\frac{1}{\omega_{z1}^2} + \frac{1}{\omega_{z2}^2} \right) + \omega_H^4 \left(\frac{1}{\omega_{z1}^2 \omega_{z2}^2} \right)}{1 + \omega_H^2 \left(\frac{1}{\omega_{p1}^2} + \frac{1}{\omega_{p2}^2} \right) + \omega_H^4 \left(\frac{1}{\omega_{p1}^2 \omega_{p2}^2} \right)} \quad (5.48)$$

$$\Rightarrow \omega_H \approx \omega_{eq} = \frac{1}{\sqrt{\frac{1}{\omega_{p1}^2} + \frac{1}{\omega_{p2}^2} - \frac{2}{\omega_{z1}^2} - \frac{2}{\omega_{z2}^2}}} \quad (5.49)$$

(5.49) ignores the higher than second order terms ω_H^4 of (5.48), which yields a good approximation if the higher frequency poles and zeros are of much higher frequencies than the ones below ω_{px} .

However, this method depends on us knowing all poles and zeros and is thus of limited use when working with an unsolved n^{th} order polynomial. Then again, one can do the slightly more crude approach to ignore higher order terms outright in the polynomial form, if they are much smaller than their predecessors, indicating that the higher order poles and zeros occur at much higher frequencies, and hopefully beyond the unity gain so that they will not have an important influence.

And maybe even better: one can use simulation tools to derive a Bode plot of a full circuit and determine the PM that way.

Part III

Digital Circuits Basics

Chapter 6

Boolean Logic

The basis of digital electronics is Boolean logic or algebra. Boolean logic describes binary valued functions operating on inputs and outputs with the values 'true' and 'false' or '0' and '1'. The fundamental operands these functions are comprised of are 'AND', 'OR', and 'NOT'. Commonly used symbols for these functions are shown in table 6.1. Their function can be expressed by **truth tables**, see table 6.2. A complete truth table for a given function lists all possible inputs together with the corresponding output.

One can combine these basic functions into longer expressions to derive more complex functions.

Just like other algebras, Boolean logic is governed by a set of rules for evaluation and manipulation of expressions. These are summarized in table 6.3.

	unambiguous	ASCII compatible
AND	$a \wedge b$	$a * b$
OR	$a \vee b$	$a + b$
NOT	$\neg a$ or $\bar{a}$	$\sim a$

Table 6.1: Symbols used for Boolean operands. a and b are Boolean variables.

a	$\bar{a}$	a	b	$a \wedge b$	a	b	$a \vee b$
0	1	0	0	0	0	0	0
0	1	0	1	0	0	1	1
1	0	1	0	0	1	0	1
		1	1	1	1	1	1

Table 6.2: Truth tables for the basic Boolean functions

$a \wedge b \vee c = (a \wedge b) \vee c$	$a \vee b \wedge c = a \vee (b \wedge c)$	(priority)
$a \wedge b = b \wedge a$	$a \vee b = b \vee a$	(commutativity)
$(a \wedge b) \wedge c = a \wedge (b \wedge c)$	$(a \vee b) \vee c = a \vee (b \vee c)$	(associativity)
$\bar{\bar{a}} = a$		(involution)
$a \wedge \bar{a} = 0$	$a \vee \bar{a} = 1$	(completeness)
$a \wedge a = a$	$a \vee a = a$	(idempotency)
$a \wedge 1 = a$	$a \vee 0 = a$	(boundedness)
$a \wedge 0 = 0$	$a \vee 1 = 1$	(boundedness)
$a \wedge (a \vee b) = a$	$a \vee (a \wedge b) = a$	(absorption)
$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$	$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$	(distributivity)
$\overline{a \vee b} = \bar{a} \wedge \bar{b}$	$\overline{a \wedge b} = \bar{a} \vee \bar{b}$	(deMorgan)

Table 6.3: Rules for Boolean function evaluation and manipulation.

Chapter 7

Combinational Logic Circuits

The digital transistor model of a voltage controlled switch can be used to implement Boolean functions. Such implementations of Boolean logic functions are referred to as **combinational logic circuits**. Basic functions in digital electronics wire diagrams can be represented as **logic gates**. Symbols for gates of the basic logic functions are shown in figure 7.1. The symbols for XOR and XNOR have also been added. Boolean variables are usually represented by voltages on electrical nodes¹ and distinguished as binary either 'true' and 'false', or respectively '1' and '0', if they are 'high' or 'low' with respect to given thresholds. Usually 'high' signals are defined as close to Vdd and above a 'high' threshold, while 'low' voltages are close to Gnd and below the 'low' threshold. The two thresholds are distinct from each other leaving a region in the middle between Vdd and Gnd, that are uncertain states. When functioning properly and given enough time to settle, digital voltage signals should not end up in this no-man's land, but when, for instance, signals are in transition or when different logic circuits with different definitions of 'high' and 'low' are improperly connected such mishaps can occur and the abstraction of the digital circuit as Boolean logic may break down. Thankfully, the fact that a whole signal range is associated with 'high' and 'low' provides usually a good error margin, and a digital gate does 'auto correct' inputs that stray from the ideal logical levels Vdd and Gnd, back to more ideal digital signals. Thus, not a single one of a CPU's billions of logic signals is misinterpreted under normal operating conditions.

7.1 Complementary PU/PD Logic

So how can we make a logic gate that implement a Boolean function? There are a few variants. Lets start with the most common one: **complementary pull-up and pull down networks**. The basic idea of a pull-down network is a network of switches/transistors between Gnd and the output node that are controlled by digital input signals at their control input/gate terminal. They are arranged in such a way to open a conductive path from the output to Gnd if, and only if, the state of the input signals form the appropriate pattern for a Boolean function to be 'false'. Examples are shown in figure 7.2. Vice versa, a

¹Representation by currents is also possible but less popular.

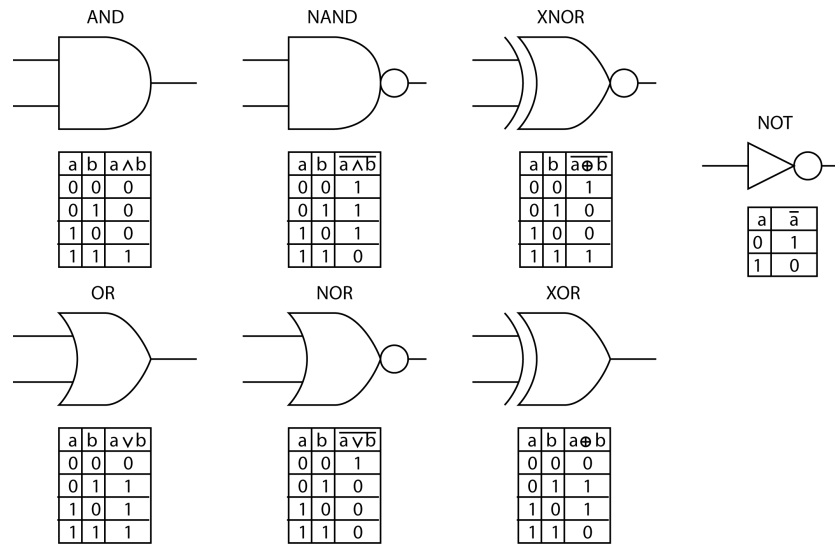


Figure 7.1: Equivalent Boolean operators, truth tables and logic gates

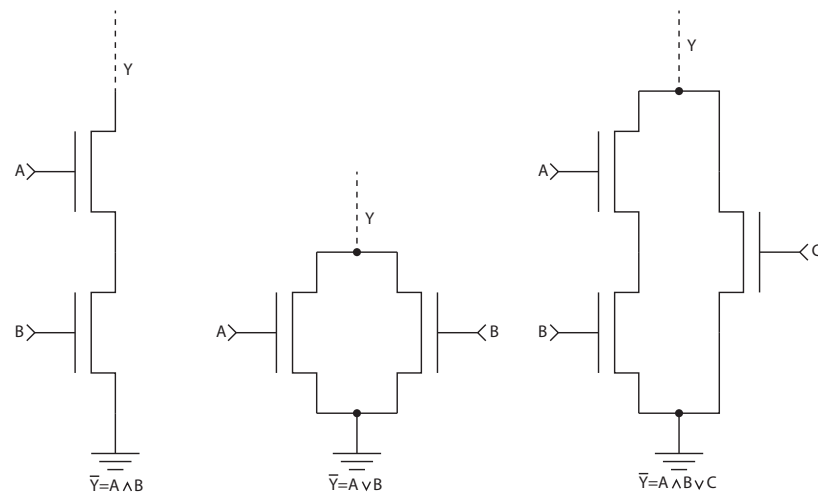


Figure 7.2: Examples of digital pull-down (PD) networks.

pull-up network forms a path only if the inputs are in a state that should make the output go 'high'.

There are a few things we should note about those PD and PU networks.

Observations:

- nFETs are efficient as switches in PD networks, and pFETs in PU networks.

This is because it is best to have the gate to source voltage large when the output undergoes a transition. So for PD nFET the source is normally constant at Gnd² and the gate to source voltage difference V_{GS} is Vdd. If a nFET is used in a PU network, its source will be pulled up in a transition, heavily reducing I_{ON} or increasing R_{ON} respectively during the transition. In fact the previous derivations of I_{ON} (4.1) and R_{ON} (4.13) were made under the assumption that the source voltage is at Gnd for nFETs and at Vdd for pFETs.

- Complex Boolean functions that would result in large PD/PU networks are often better implemented with a series of logic gates, rather than a single gate with large PD/PU networks.

As we will see shortly, large networks that require transistors in series will increase the propagation delay of a gate AND reduce the quality of the output signal, such that an equivalent implementation with multiple gates in series can oftentimes achieve an overall better performance.

- nFET PD networks can only implement particular Boolean functions, i.e. those that have an equivalent expression where the output being 'low' is a function of non-inverted input variables, e.g. $\bar{Y} = A \vee B \wedge C$. Its the other way round for PU networks implemented with pFETs, i.e. the output being 'high' as a function of all inverted input variables, like $Y = \bar{A} \wedge (\bar{B} \vee \bar{C})$. Note in particular, that an AND or an OR gate does *not* fit that requirement! Thus, it is the NAND and the NOR gate that can directly be implemented as a single stage CMOS logic gate, while AND and OR are composed of two stages, e.g. a NAND followed by an inverter or a NOR followed by an inverter.

One can complement a pull-down network simply with a *pull-up resistor* towards Vdd (figure 7.3 A). That resistor maintains thus the default of the output being high unless the inputs make the pull down network conductive. This is a possible implementation with some drawbacks. For one, while the output is being pulled low, this circuit consumes a constant current. Furthermore, the resistor needs to be dimensioned correctly such that the PD network in fact is able to pull the output voltage down. Also, actual resistors are costly in layout space in integrated circuits, much more so than transistors.

The latter can be amended by replacing the resistor with a pull up *transistor*, i.e. a pFET transistor with an appropriate constant gate voltage and of appropriate dimensions (figure 7.3 B). For example one can choose a pFET with its gate voltage at Gnd, which then needs to have a quite small $\frac{W}{L}$ ratio

²When multiple nFETs are in series this might not be true for nFETs that are not directly connected to the Gnd terminal at the start of the transition. However, if so their source terminal will rapidly be depleted of positive charge and be pulled to Gnd efficiently by the nFET below them.

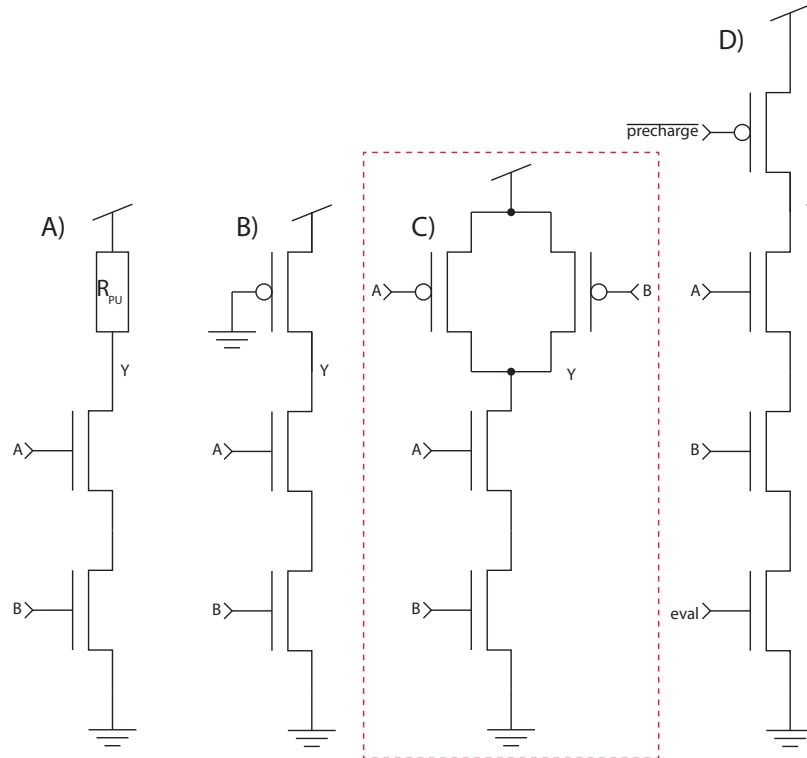


Figure 7.3: Ways of implementing a NAND gate with a nFET pull-down network. The most widely applied is variant C) the **complementary pull-up/pull-down network** version in the red dashed box. A) and B) use a static pull-up, with respectively a resistor and an open pFET. They require that the pull up strength is weak enough for the pull-down network to be able to overcome. D) is a **dynamic logic** implementation. 'precharge' and 'eval' may simply be the same shorted clock signal, or non-overlapping clock signals. The output is only valid after 'eval' has gone high (see section 7.4)

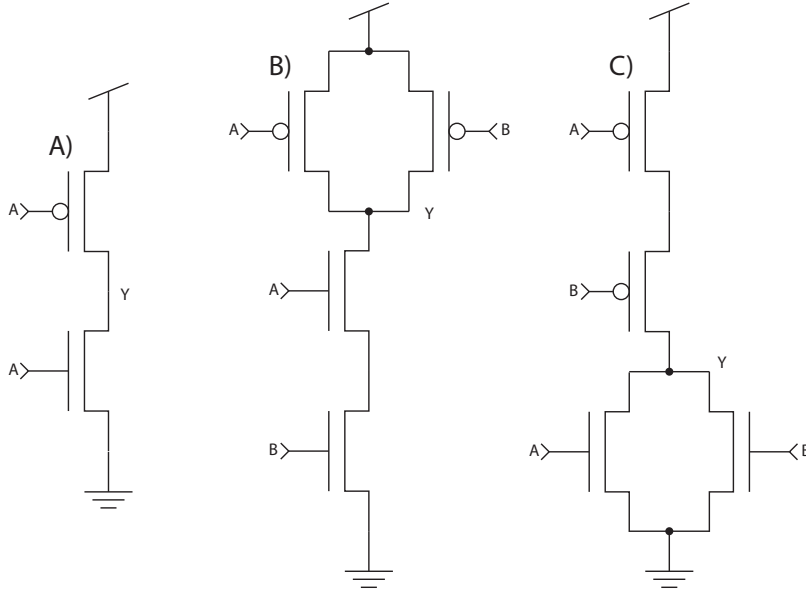


Figure 7.4: Boolean operands that are most easily implemented as logic circuits with complementary PU/PD networks, here with complementary PU/PD networks: A) NOT or inverter, B) 2-port NAND, C) 2-port NOR

to make its static R_{ON} big enough for the active PD network to be able to pull the output low. So this solves the layout space problem, but not the constant current problem for maintaining a 'low' state at the output.

Thus, the normal solution here is to use *both* a PU *and* a complementary PD network that implement the *same* logic function, i.e. when the PD network is active the PU network is inactive and vice versa (figure 7.3 C)). This is the most prominent solution used in static logic circuits. It does not consume any current to maintain a state (other than leakage!), and is only using small transistors, i.e. it is not excessive in layout space. Figure 7.4 shows the basic Boolean operands' implementation as static logic gates with complementary PU/PD networks.

Figure 7.3 D) shows a **dynamic logic** inverter. We will get back to that variant in section 7.4.

7.1.1 Balancing PU and PD networks

Let us have a look at the propagation delays of complementary PU/PD static logic. In section 4.2 we have had a first look at the propagation delay t_p when a single transistor pulls a digital signal up from Gnd to $\frac{V_{dd}}{2}$ or down from Vdd to $\frac{V_{dd}}{2}$. Obviously, logic gates need to do both, so let's distinguish the two by writing t_{pHL} and t_{pLH} for the delay in pulling down and up respectively, and let us redefine t_p as the *average* of the two:

$$t_p = \frac{t_{pHL} + t_{pLH}}{2} \quad (7.1)$$

We shall use the resistive model of a switched transistor derived in sub-

section 4.2.2 and equation (4.13) to gauge propagation delays. They are thus proportional to $R_{ON}C_L^*$ and consequently inversely proportional to parameter $k = \mu C_{OX} \frac{W}{L}$, as well as proportional to the total load capacitance C_L^* of the output node. Note here that the carrier mobility μ is usually 2 to 3 times bigger for nFETs than for pFETs.

C_L^* at the output of a gate can be dissected into three main contributors: the capacitance at the output of that gate alone C_O , the input capacitance C_I of anything it might be connected to, e.g. typically other logic gates, and parasitic capacitances summarized as C_W caused by interconnect of the two, in particular if they are connected by a long cable.

$$C_L^* = C_O + C_W + C_I \quad (7.2)$$

7.1.2 Inverter

let's first consider the simplest case of an inverter connected to another inverter. Then the main components of C_O are:

$$C_O = 3C_{GDp} + 3C_{GDn} + C_{DBp} + C_{DBn} \quad (7.3)$$

The curious factor 3 for the two C_{GD} is explained by a Miller effect, i.e. the feed forward charge injection one gets from these capacitors with a perfect step input. See the next subsection 7.1.3.

and C_I is:

$$C_I = C_{GSp} + C_{GSn} + C_{GDp} + C_{GDn} \quad (7.4)$$

Be reminded here that all of these capacitors do scale with the width W of the transistors, and C_{GS} does *also* scale with the length L . (See equations (2.26), (2.27), (2.29) and (2.28)).

Considering the respective $\mu_n C_{OX}$ and $\mu_p C_{OX}$ as given for now, one can contemplate to keep $L_{n/p}$ at their minimum³ and to increase $W_{n/p}$ to reduce R_{ON} , but this will proportionally increase both C_{OUT} and C_{IN} as well. It will *still* reduce t_p though, as it will diminish the relative contribution of C_W . However, this is only worthwhile up to a point where the influence of C_W becomes insignificant. Actually mostly W is kept close to a minimum too, except for logic gates that need to drive long cables or chip pads that are used as connections off-chip. This is because layout space is also a costly commodity and generally it is desirable to cram as many logical gates into an as small as possible space.

Let us first consider the simplest of all digital gates, the inverter, consisting of just one nFET and one pFET (see figure 7.4). R_{ON} is thus derived for a single transistor in each direction (i.e. PD and PU) according to equation (4.13). Considering interconnect capacitance C_W to be negligible, using minimum size transistors will result in a minimal C_L^* . Further assuming that the minimum transistor dimensions are the same for nFET and pFET (which is usually the case) we get asymmetric propagation delays if $\mu_n > \mu_p$ (which is also usually the case), i.e. $t_{pLH} > t_{pHL}$. This has a number of undesirable side effects:

³Note that the name for a given technology actually is based on this minimum transistor length, e.g. Intel 7nm CMOS technology or the like.

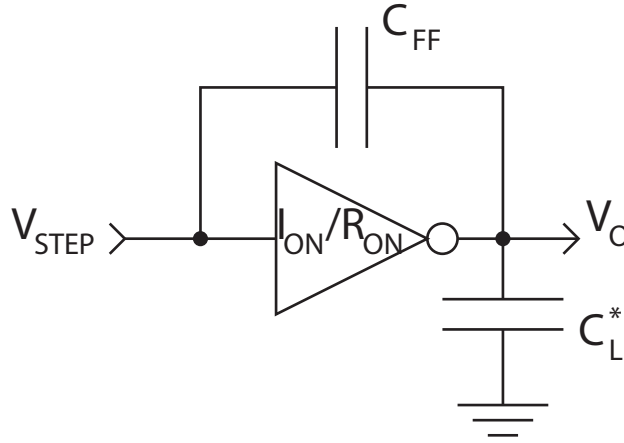


Figure 7.5: Illustration of the Miller effect affecting the output node capacitance of an inverter. Here, the inverter symbol itself does only represent the ideal switches initiating the pull up or pull down via either the current source- or resistor model. C_{FF} represents the capacitors C_{GD} between input and output.

- The real switching threshold will be skewed: $V_{sw} < \frac{V_{dd}}{2}$. Thus, the error margin of a 'low' input signal to be incorrectly identified as 'high' is reduced.
- The worst case propagation delay is much worse than the average t_p . Thus, the clock cycle needs to be slowed down to be able to accommodate that worst case, i.e. slow enough that one can be sure that the logic in question can have a stable output within the clock cycle.

Thus, it is desirable to balance the t_{pLH} and t_{pHL} . This is done by making pFETs wider than nFETs such that $\mu_n C_{OX} \frac{W_n}{L_n} \approx \mu_p C_{OX} \frac{W_p}{L_p}$. So if $\mu_n = x\mu_p$ (where usually $2 \lesssim x \lesssim 3$) we choose for the inverter $W_p = xW_n$. Let us denote the transistor dimensions for a *balanced* inverter as $p = \frac{W_p}{L_p} = \frac{xW_n}{L_p}$ and $n = \frac{W_n}{L_n}$ (Fig. 7.6 A))

7.1.3 Inverter: Miller approximation of C_O for step input

In digital electronics the feed forward effect of the 'Miller capacitances' between input and inverted output node(s) (e.g. C_{GD} for inverters) is taken into account for approximating capacitance at the output node (in contrast to analog where one is more interested in the Miller effect on the *input* of a transconductance amplifier and its BW), where one assumes a perfect step at the input. For the inverter for instance (see figure 7.5), if we assume the input step to be from Gnd to Vdd, and are interested in the time t_p it takes for the output to go from Vdd to $\frac{V_{dd}}{2}$. Then the two C_{GD} become important as a feed forward capacitance, so let's now call their sum C_{FF} . It will inject a charge $Q_{FF} = C_{FF} * V_{dd}$. Let's call the total capacitance at the output other than that feedback C_L^* . Then the

total charge that needs to be drawn from the output node to move from Vdd to $\frac{V_{dd}}{2}$ is:

$$Q_{V_{dd} \rightarrow \frac{V_{dd}}{2}} = \frac{V_{dd}}{2} (C_L^* + C_{FF}) + C_{FF} * V_{dd} = \frac{V_{dd}}{2} (C_L^* + 3C_{FF}) \quad (7.5)$$

If that charge is supplied by a current source I_{ON} then the time to reach $\frac{V_{dd}}{2}$ (i.e. the propagation delay t_p) is:

$$t_p = \frac{Q_{V_{dd} \rightarrow \frac{V_{dd}}{2}}}{I_{ON}} = \frac{V_{dd}}{2} \frac{C_L^* + 3C_{FF}}{I_{ON}} \quad (7.6)$$

So it's perfectly equivalent to increasing the effect of C_{FF} threefold.

Oftentimes, the same assumption is also made when one uses the model of R_{ON} instead of I_{ON} , resulting in:

$$t_p = \ln(2) R_{ON} (C_L^* + 3C_{FF}) = 0.69 R_{ON} (C_L^* + 3C_{FF}) \quad (7.7)$$

even though this is not correct: the real time constant in that case is still just $R_L^* (C_L^* + C_{FF})$ but the 'starting point' V_{start} is now higher than Vdd since the step charge injection raises the voltage to V_{start} .

$$V_{start} = V_{dd} \left(1 + \frac{C_{FF}}{C_{FF} + C_L^*} \right) \quad (7.8)$$

to get t_p one needs to solve

$$e^{-\frac{t_p}{R_L^* (C_L^* + C_{FF})}} = \frac{\frac{V_{dd}}{2}}{V_{start}} \quad (7.9)$$

$$= \frac{1}{2 \left(1 + \frac{C_{FF}}{C_{FF} + C_L^*} \right)} \quad (7.10)$$

$$= \frac{1}{2} \frac{C_{FF} + C_L^*}{2C_{FF} + C_L^*} \quad (7.11)$$

$$t_p = R_L^* (C_L^* + C_{FF}) \left(\ln 2 + \ln \frac{2C_{FF} + C_L^*}{C_{FF} + C_L^*} \right) \quad (7.12)$$

Be that as it may: no-one uses that, but one assumes simply the contribution of $3C_{FF}$ (actually many books use $2C_{FF}$) instead into the already heavily simplified model of R_{ON} , i.e. one uses (7.7). Qualitatively the effect is the same, i.e. that the propagation delay t_p is made slightly longer.

7.1.4 More complex gates

More complex logic gates may have multiple paths by which the output may be pulled up or pulled down. Figure 7.6 shows some examples. Using the resistive model R_{ON} , the total resistance is derived from computing the total R_{ON} of the active PU or PD network. Note that there might be more than just one case to consider, as there might be multiple possible PU or PD paths dependent on how all the inputs switch. In that case it's the worst case scenario that is most important. It may not be possible to balance the transition of the output

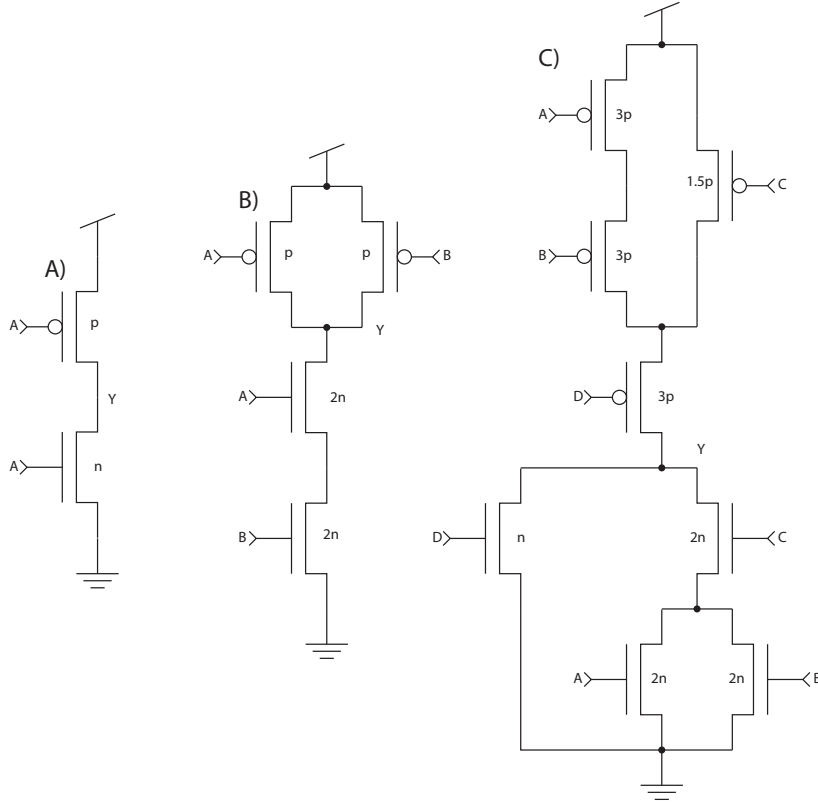


Figure 7.6: Illustrating balancing logic gates for symmetric worst case t_{pHL} and t_{pLH} . n and p indicate the balanced inverter's respective transistor width to length ratios resulting in equivalent R_{ON} for both PU and PD transistors. The ratio $\frac{W}{L}$ needs to be increased when multiple transistors are in series if one wants to maintain the same total worst case R_{ON} for more complex gates. The two examples are a NAND ($\bar{Y} = A \wedge B$, subfigure B)), and the function $\bar{Y} = D \vee (C \wedge (A \vee B))$ (subfigure C)). Note that while the worst case R_{ON} is the same in all three instances, still C_O at the output node Y is increased for B) and C) due to the wider and more numerous transistors connected to it. Furthermore there are now 'internal' nodes that also may need to be charged during a transition! So B) and C) are balanced but still have longer propagation delay t_p .

for *every* possible transition combination of switching inputs. Instead, we shall aim at balancing the *worst case* transition between the PU and PD network. In the figure 7.6 each transistor is marked with a factor xp or xn indicating its respective $\frac{W}{L}$ ratio, p and n being the $\frac{W}{L}$ of a balanced inverter. R_{ON} for each transistor is proportional to $\frac{L}{W}$. If one needs to make sure that the sum of all transistors' R_{ON} in series along each possible path is equal to a single transistor of a balanced inverter, then the total worst case resistance is equal to that of the balanced inverter. NOTE: this does *only* take care of balancing R_{ON} and making it no bigger than in the case of the inverter. However, C_{OUT} will still increase (and there will be some 'internal' nodes in the gate with node capacitances that may also need charging/discharging), thus the delay proportional to $R_{ON}C_L$ will still increased relative to an inverter!

Observations:

- 'NAND is better than NOR for layout space', that is to say, it's preferable to have the longest serial branch of a gate in the PD network rather than the PU network. That is because $p > n$, so a relative increase in the transistor width of a pFET consumes more space than the same relative increase of an nFET's dimensions.
- Single stage gates with large PU/PD networks are often worse for propagation delay than an equivalent multi-gate implementation. Comparing the propagation delay of figures 7.6 B) and C) with just the RC model is quite inaccurate, even if we would extend it to include the internal nodes' capacitances. Just the capacitance at the output C_{OUT} is 1.5 times bigger for C) than for B), so that would only make it 1.5 time slower, but the internal capacitances contain also source to gate capacitances that are way bigger than just the drain to gate capacitances of the inverter. So at that point the use of a simulation tool is recommended to gauge propagation delay more accurately and to compare a multi stage implementation of C) to the single stage shown here.

7.2 Dynamic and static power consumption

With today's enormous transistor density in CMOS technology, power consumption and heat dissipation are major challenges. Even if each logic gate dissipates only a weeny tiny bit of power, if you amass several billions of them on just a few square centimetres, they will consume a lot of energy and turn it into concentrated heat. Cooling of CPUs has become a real challenge. The power consumption of digital circuits is often divided into two categories: **dynamic power consumption** that results from the gates actively computing and thus from the energy consumed when digital signals are switching states, and **static power consumption** that is consumed irrespective of activity, i.e. also when the circuit is idle, and consists mainly of leakage currents of PN junctions and also through transistors that are 'off'. For an inverter, the major contributors for dynamic switching power are a) the charge/energy needed to affect a state change of the output and that energy is proportional to $V_{dd} * C_O$, while the power is proportional to $V_{dd}^2 * C_O$ at the same clock rate, and b) a so called **cross current** flowing directly from the inverter's supply/ V_{dd} to Gnd that occurs in the transition of the input signal (see figure 7.7). b) occurs while the

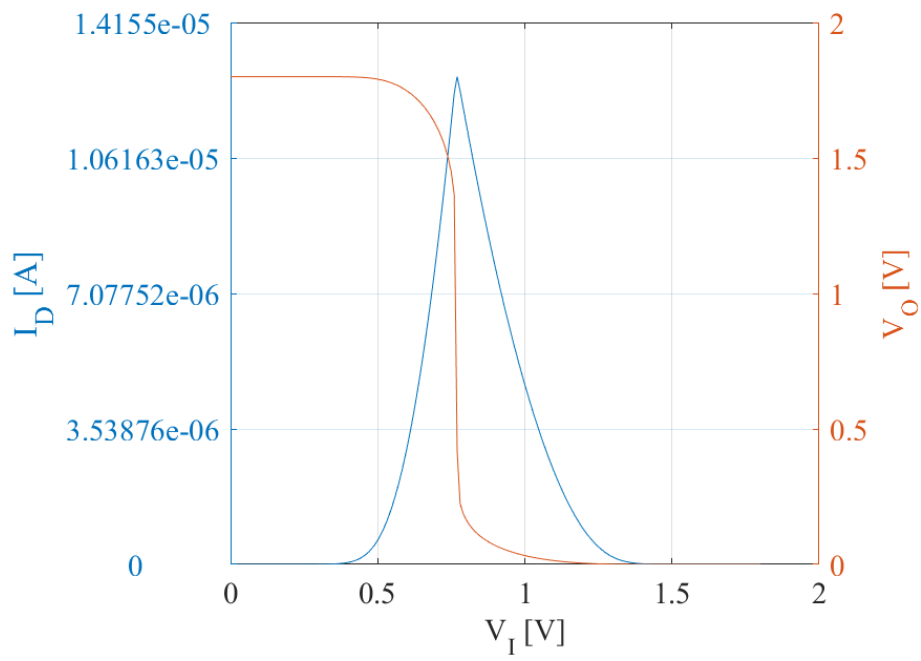


Figure 7.7: Illustration of inverter cross current for a 180nm CMOS technology with 1.8V V_{dd} and the pFET 3 times the width of the nFET.

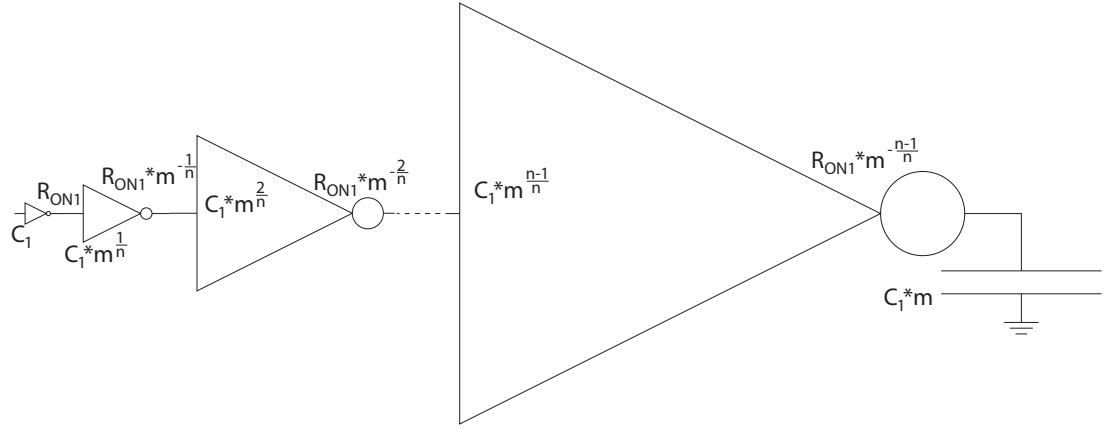


Figure 7.8: Illustrating a cascade of ever bigger inverters for driving a large capacitive load, e.g. when connecting to an off-chip element. Ideally, each stage is e times bigger (that is a wider W with the same length L for the two transistors) than the previous for minimal propagation delay and there should be $\ln m$ stages. However, m is often times unknown, so practically 2 to 4 inverters with a size increase of 2-4 per stage are often used and will pragmatically improve the propagation delay, if not perfectly optimize it.

(not infinitely fast) input is somewhere between V_{dd} and Gnd and is roughly worst while at $\frac{V_{dd}}{2}$. That worst current is proportional to V_{dd}^2 when it's in strong inversion and to $e^{\frac{V_{dd}}{U_T}}$ when in weak inversion.

7.3 Driving large external loads C_W

Microscopic integrated circuits do sometimes need to drive some macroscopic loads, typically when driving bus lines or most severely when connected to a pad on the chip that in turn is connected via a micro-wire to a conductor on a PCB. In that case one should employ a gigantic inverter (i.e. with large $\frac{W}{L}$ transistors) as a buffer between the internal microscopic world of microscopic capacitances and the macroscopic world of macroscopic capacitances of the PCB. However, this is just dumping the problem onto the digital gate just *before* that inverter that now has to drive an unusually big input capacitance of this humongous inverter.

The best solution to this is actually a cascade of inverters with ever bigger $\frac{W}{L}$ ratio (figure 7.8). If these larger $\frac{W}{L}$ ratios are implemented by increasing W while leaving L constant, and if one considers the load of each of those inverters to be dominated by the input capacitance of the next stage, and if an inverter connected to a same size inverter has a gate delay t_{p1} , then an inverter connected to an x times larger inverter has a propagation delay of

$$t_{px} = xt_{p1} \quad (7.13)$$

So if $C_{W_{ex}}$ is the external load after the last stage and it is m times bigger

than C_1 which is the input capacitance of a 'normal' sized inverter as used in on-chip logic, then the total delay of such an inverter driving $C_{W_{ex}}$ directly is:

$$t_{p_{tot}} = mt_{p_1} \quad (7.14)$$

If one inserts an extra buffer inverter of a size $\sqrt{m}$ times bigger than a 'normal' inverter, that consequently also has a R_{ON} that is $\frac{1}{\sqrt{m}}$ that of a normal sized inverter R_{ON_1} and an input capacitance that is $\sqrt{m}C_1$. So the total delay is now:

$$t_{p_{tot}} = \sqrt{m}t_{p_1} + \sqrt{m}t_{p_1} = 2\sqrt{m}t_{p_1} \quad (7.15)$$

And more generally for $n - 1$ inverters beyond the first with the same ratio of increased size between each consecutive stage one gets:

$$t_{p_{tot}} = nm^{\frac{1}{n}}t_{p_1} \quad (7.16)$$

And it turns out (i.e. it can be shown mathematically, but not here) that the optimal factor of increase between the stages $m^{\frac{1}{n}}$ resulting in the minimal delay is e :

$$m^{\frac{1}{n_{opt}}} = e = 2.7183 \quad (7.17)$$

So the optimal number of stages n_{opt} is:

$$n_{opt} = \ln m \quad (7.18)$$

Practically, one uses an increase between stages of 2 to 4 for good results, but since m is often unknown or might vary, as the ASIC is connected to different loads when used, it's not clear a priori how many stages n that are needed. Anyway, any factor $m^{\frac{1}{n}}$, even when > 4 and any number of stages are better than none, in practice.

7.4 Dynamic logic

Dynamic logic circuits (figure 7.3 D)) are not widely used and are finicky, but have their place where ultimate speed is required.

They implement only either a switched PU or PD network, just like variants A) and B) in figure 7.3, but as opposed to A) And B) that use a static PD/PU resistor or transistor on the opposing side, they use switches to toggle between the logic PU/PD network and the opposing PD/PU 'reset', i.e. they toggle between a reset/**precharge phase** where the output is set to a default state, and an **evaluation phase** where the logic function is evaluated and determines whether the default state is maintained or switched. The term *dynamic* results from the fact that the 'default' state resulting from the precharge phase is not statically/actively maintained during the evaluation phase if the PD/PU logic does not switch it. Instead, it is only passively stored as a charge on the output node capacitance. Consequently, leakage currents can destroy that state over time if it is not refreshed and/or 'used' quickly enough. Also all kinds of noise can affect it, e.g. coupling noise from nearby switching signals. Furthermore, if the logic PD/PU network is not turned on completely and no complete path

exists to respectively Gnd/Vdd, but if some individual transistors in the network *are* switched on, limited charge from internal nodes between transistors may be dumped on the output as well and might distort the dynamic voltage on the output node. So in conclusion: if you do not *have* to use dynamic logic, it's better to avoid it.

Figure 7.3 D) shows an implementation of a NAND gate. Note that it is advantageous to implement the switched PD network rather than the PU network, since pFETs are usually chosen to be wider and thus need more space. Here, only a single pFET is needed instead of the two for a static logic NAND gate with complementary PU and PD, but an extra nFET is needed to switch the PD network. So again a total of four transistors, but now more nFETs which oftentimes are chosen smaller.

Pros and Cons:

- + Fewer transistors for gates more complex than a NAND or NOR, and fewer pFETs already for NAND and NOR with switched PD network. (good for layout density)
- + With non-overlapping precharge and evaluation phases there is no cross current (good for power)
- + Less parasitics contributing to C_{OUT} and typically only one C_{GS} contributing to C_{IN} instead of two (good for speed and power)
- Always switching even when evaluation state stays the same (bad for power)
- Sensitive to noise and timing due to 'passive' maintaining of state (bad for design effort)
- Can consequently generate errors when run too slowly or due to noise (e.g. cross talk) in evaluation phase

Let us look at an example in figure 7.9 (a two port multiplexer). Note that for simplicity sake, we use a single CLK signal instead of 'eval' and 'precharge', but remember that to avoid cross currents one needs to use two non-overlapping signals for the evaluation footer and precharge header.

The logic gate representation of the multiplexer is shown in the upper right. The function of a multiplexer is to route one of multiple digital signals to an output. Here the multiplexer chooses between signals A and B according to signal S. If S is high, O will be the same as A. If S is low, O will be the same as B. The circuit is implemented with three NAND gates. There is a potential problem when using the same CLK/eval signal for the consecutive stages: If the evaluation phase starts simultaneously for the second stage, it may be that the first stage outputs are still in their precharge state and only beginning to be pulled low if their evaluation state is supposed to be low. Thus the PD network in the second stage is open and may start to pull the output low by mistake. To avoid this, the evaluation phase of the second stage *needs* to be delayed and for any consecutive stages the CLK/eval signal needs to be **staggered** this way!!! One can, for example, run the CLK signal through a double (static) inverter before it is applied to the next stage and one needs to make very sure that this delay is sufficient: inverters may be faster than the

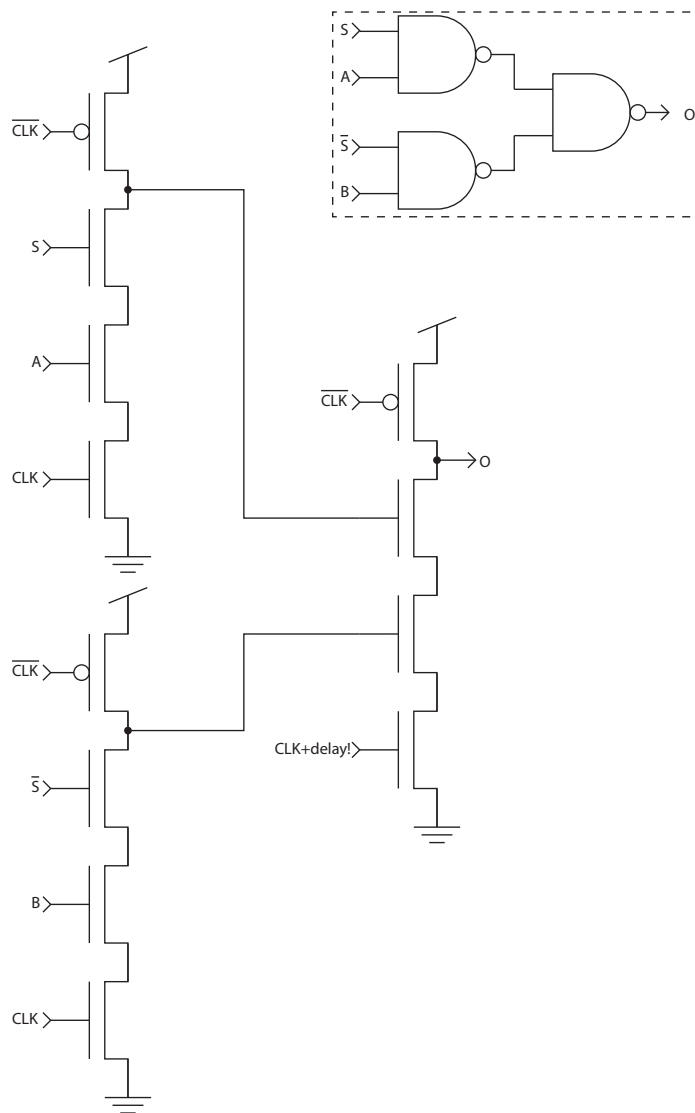


Figure 7.9: Generic dynamic logic implementation of a digital multiplexer

NANDs here, so the double inverter adds some safety margin. But note that there always is mismatch/variability between CMOS elements after production and with billions of such gates a very unlucky outcome for one of them can never be ruled out entirely.

Thus, there is a dynamic logic variant that avoids this particular problem and the need for clock staggering entirely, called **domino logic** (Fig. 7.10): it makes sure that the default precharge state seen by the next stage is *low* if that stage is employing a PD logic network in the evaluation phase. Thus this next stage cannot start to pull its output low prematurely. This can be achieved in either of two ways. The first is to still employ only stages with PD networks but to interpose a static inverter between stages. The multiplexer circuit can thus be implemented like shown in figure 7.10. Now the default precharge signals seen by the second stage are low. Consequently the PD network cannot start pulling low prematurely. Only if the first stage has completed an eventual change of its default stage, can a change in the next stage occur. In fact, the switches in consecutive stages initiating the evaluation phase can be omitted entirely, i.e. the nFET receiving the CLK signal, and the correct function is maintained. This has been done in the illustration. *However*, this invites a significant cross current during the pre-charge phase, i.e. when the pre-charge pFET starts conducting and the inputs from previous stages are still in their evaluation phase for a split-(pico)second!!! Thus, it may be worth keeping the evaluation switch and a non-overlapping CLK signal also for consecutive stages. Also if the stage does receive inputs from static logic, i.e. not the appropriate domino logic circuits, the evaluation phase switch/nFET needs to be kept. So typically this might be the case for a first stage that interfaces to static logic. In the illustration, the two gates in the first stage have kept their evaluation switch.

Note that due to the extra inverter, this solution requires more transistors and becomes slower. Also, the generic smallest building blocks are now ORs and ANDs rather than NORs and NANDs, as reflected in the gate representation of the MUX in the figure. Here, the latter is no big impediment, as there is only three gates needed in either case.

A variation of the domino logic is the so called **NP domino logic** that avoids the inverters and thus the extra propagation delay and extra transistors. The multiplexer implementation is shown in figure 7.11. Here, consecutive stages are implemented toggling between PD and PU networks for the logic function. Thus the default precharge state of a PD network gate is high, and if connected to a PU network gate will prevent premature pull up of its output signal. The same domino effect results from this without the inverters. A disadvantage layout-wise, is that PU networks require the usually bigger pFETs. So the layout space requirement will be bigger as opposed to the generic dynamic logic implementation in figure 7.9.

So dynamic logic is typically employed when extremely high speed is required. Another application area is in asynchronous circuits. Remember that the length of the clock cycle in synchronous circuit is determined by the worst case propagation delay of combinational logic executing a computation where the result is needed at the end of the clock cycle. **Dual rail Domino logic** can implement a different kind of combinational logic that can actually *signal* once it has finished a computation, as illustrated in figure 7.12. Thus, rather than using a clock that is determined by the worst case delay, a combinational logic

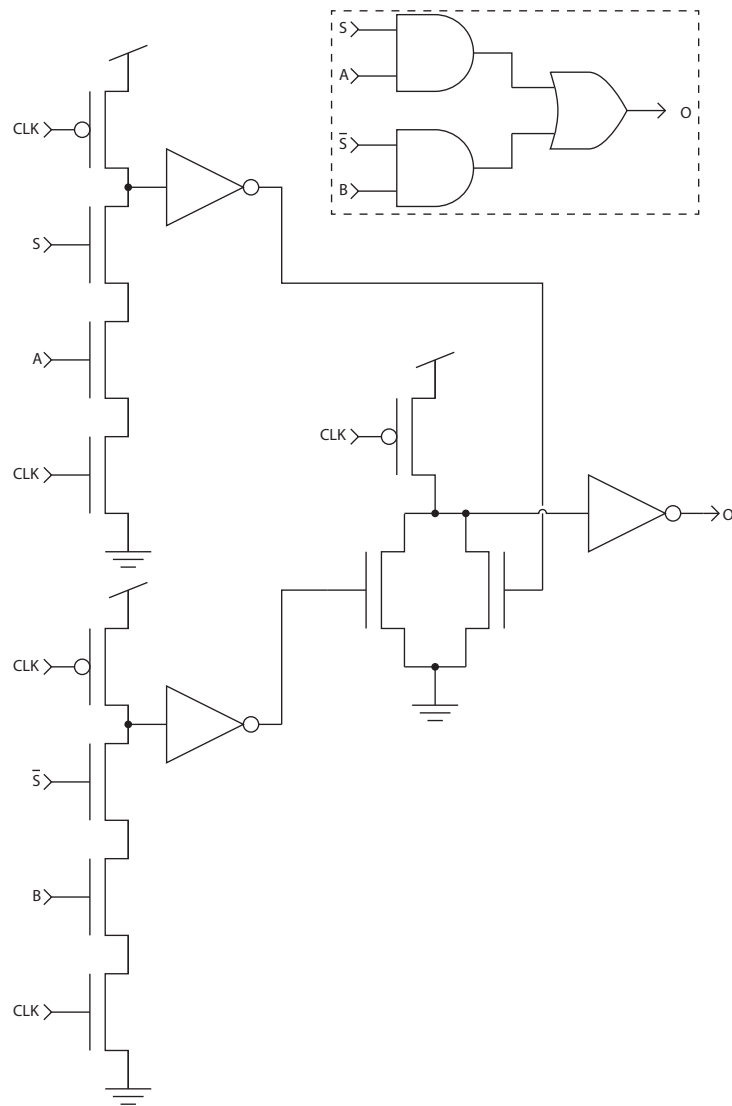


Figure 7.10: Domino logic implementation of a digital multiplexer

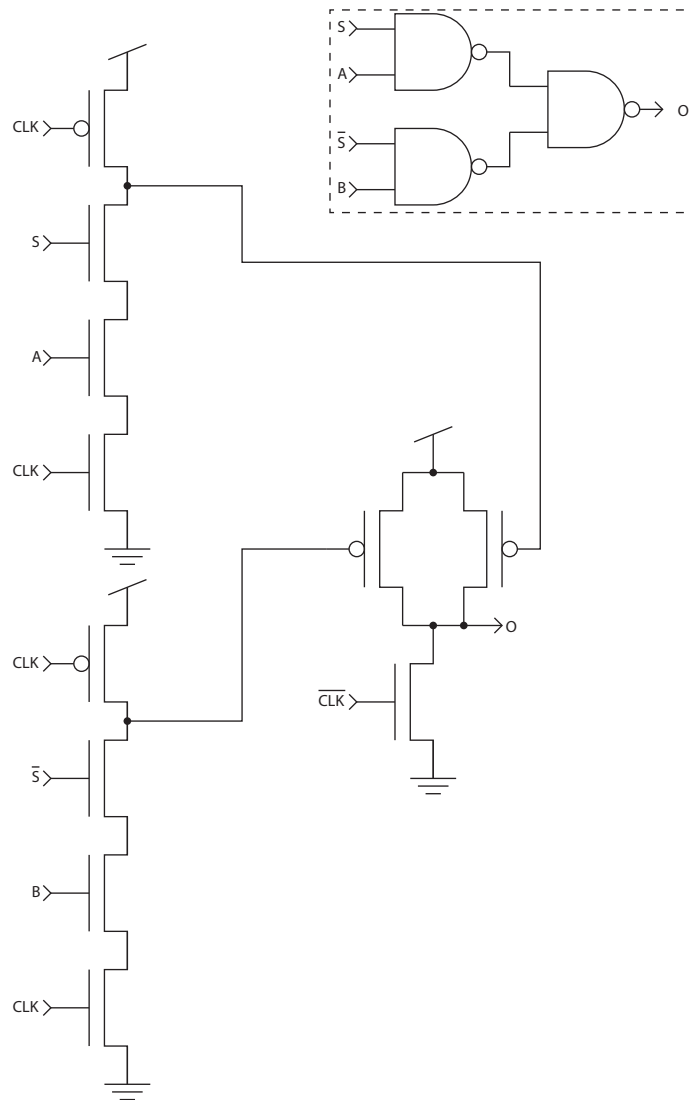


Figure 7.11: NP domino logic implementation of a digital multiplexer

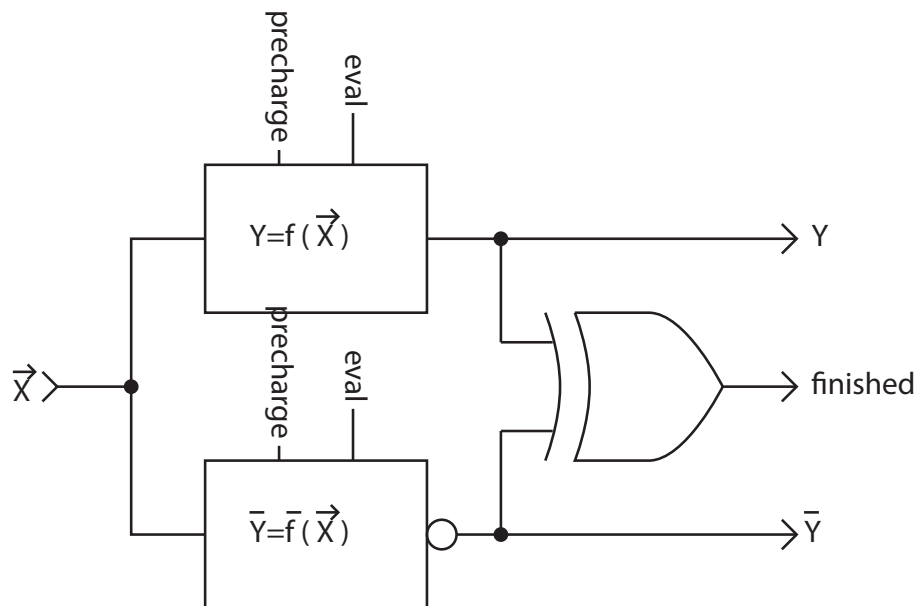


Figure 7.12: Dual rail domino logic for asynchronous circuits where it can be determined when the combinational computation is finished in the evaluation phase. In the precharge phase we have $\bar{Y} = Y$ and in the evaluation phase the computation is finished and the output is valid as soon as $\bar{Y} \neq Y$.

block can simply signal the next stage that the result is ready, and a sequence of processing steps can be executed as fast as possible, i.e. not as slow as the worst case would require.

7.5 Example of advanced combinational logic circuits

Just briefly mentioned:

- decoders
We are not thinking about video/audio decoders here but the simplest form of decoder that receives a binary number as input and turns it into a 'one-hot' code, so for example 3 bit binary 5 '101' into 8 bits where the $(5+1)^{th}$ (so the 5^{th} when starting at '0') bit is active and all others are inactive, here counting from the right: '00100000'
- encoders
These would implement the inverse function of decoders, i.e. turn a one-hot code into a binary number, e.g. '01000000' to '110' (7).
- multiplexers
Those 'route' a single bit out of N bits to a single bit output. A binary number as control input identifies which of the N input bits should be routed to the output. So the multiplexer takes two groups of bits as

inputs, the bits among which a single bit is selected and the bits forming the binary number that selects said bit. (There exist analog multiplexers as well that rout one of many analog signals, but that's a 'mixed-signal' circuit and not purely combinational logic.)

- demultiplexers

The inverse of multiplexers: a single input bit gets routed to a specific output bit among an array of output bits. Again a control input identifies the selected output bit.

- Arithmetic logic unit (ALU)

A core component of any processor: typically it takes as input two binary numbers (can also be seen as bit arrays) and as a control signal a so called 'instruction'. Based on the instruction it can execute a number of operations on the two input finally producing a single output binary number. In this simplest form it is but a combinational circuit that takes but one clock cycle to execute its function, but it can also become more complex where some instructions are executed in multiple clock cycles, and thus it would not be a combinational circuit anymore, but a sequential circuit (see next section!).

Chapter 8

Sequential Logic Circuits

Combinational logic circuits as discussed so far implement Boolean functions. That is to say, combinational logic does not contain any form of memory or feedback connections and the input defines the output completely. In contrast, **sequential logic circuits** do contain feedback loops and thus memory elements and an internal state such that the output does not only depend on the input, but also on the internal states and thus the history or sequence of inputs.

So a truth table describing a sequential logic circuit will not just describe the output for all possible inputs, but will need to describe the outputs for all possible inputs, *as well* as for all possible internal states and the former state of the output. Such a truth table is called a **characteristic table**.

Note that by introducing feedback loops one can (inadvertently or on purpose) also design circuits that oscillate or end up in a **metastable state** (i.e. where some signals end up not being 'digital' anymore but get stuck at a voltage between Gnd and Vdd), if there is a circular loop of digital signals that cannot achieve a consistent stable state. The typical example would be an inverter with its output shorted to its input. It will either settle on a non-digital voltage somewhere close to $\frac{V_{dd}}{2}$ or oscillate wildly. To avoid such a fate, oftentimes short feedback loops are only employed within specifically designed and well understood memory elements.

8.1 Asynchronous latches

A simple useful sequential logic circuit is a 1-bit storage element. One quite straight forward implementation is *two* inverters connected in a loop (figure 8.1 A)). This loop will indefinitely maintain one of two possible states, either $Q = 1$ or $Q = 0$. In order to be able to control the stored state figure 8.1 B) introduces a 'gating' control signal E . If $E = 0$, the feedback loop is broken and the output Q adapts the state of the input signal D . This storage element is known as **D-latch** or **transparent latch**. Its characteristic table is given in table 8.1.

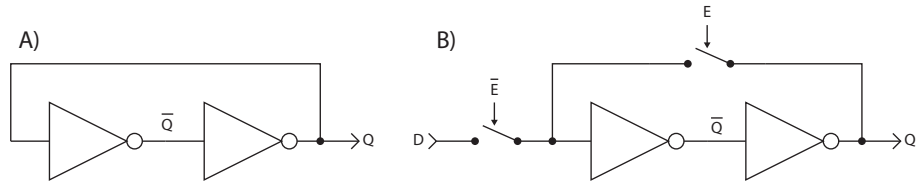


Figure 8.1: A) principle of a double inverter loop storage cell. State Q can be 0 or 1 and is maintained indefinitely, without input to change it. B) The D-latch or transparent latch rectifies this with two inputs: when $E = 1$ then the state is maintained, and when $E = 0$ the latch becomes a 'transparent' double inverter that just conveys the input D .

D	E	Q	$\bar{Q}$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

E	Q
0	D
1	Q

Table 8.1: D-latch characteristic table, with full explicit table on the left and an abbreviated version on the right.

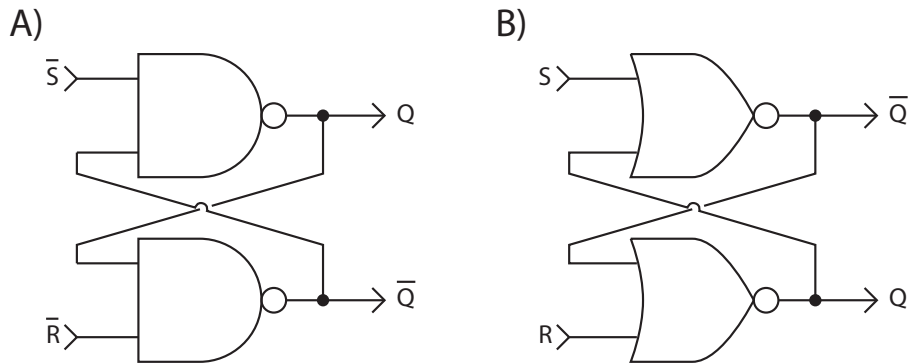


Figure 8.2: Another popular implementation of a latch is the RS-latch. When S is active and L is inactive, $Q \leq 1$ and $\bar{Q} \leq 0$ and vice versa. When both S and R are inactive, the state is maintained. When both R AND S are active that is actually 'illegal' and the state in a general definition of a RS-latch is not defined. For the specific implementations A) both $Q \leq 1$ AND $\bar{Q} \leq 1$ and for B) it's $Q \leq 0$ AND $\bar{Q} \leq 0$.

R	S	Q	Q	$\overline{Q}$
0	0	0	0	1
0	0	1	1	0
0	1	0	1	0
0	1	1	1	0
1	0	0	0	1
1	0	1	0	1
1	1	0	0	0
1	1	0	0	0

R	S	Q	$\overline{Q}$
0	0	Q	$\overline{Q}$
0	1	1	0
1	0	0	1
1	1	0	0

Table 8.2: NOR gate implementation of RS-latch characteristic table, with full explicit table on the left and an abbreviated version on the right.

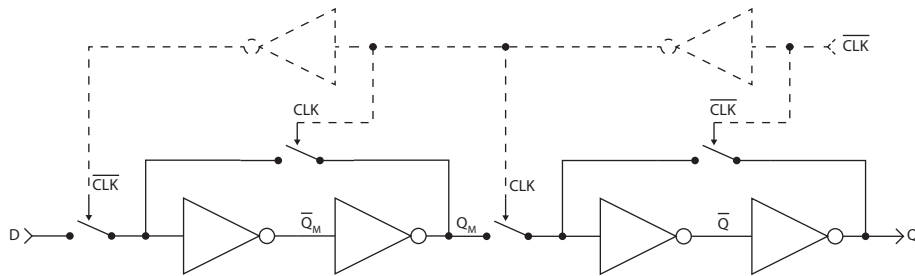


Figure 8.3: D-flipflop implemented with master slave D-latch pair. The inverters on top are not strictly necessary, but ensure a safe timing of CLK and $\overline{CLK}$: it's important that the right hand D-latch latches *before or simultaneously with* the left hand side latch going transparent, i.e. if the left hand side becomes transparent *first* it could lead to a signal feed through at the negative flank of CLK .

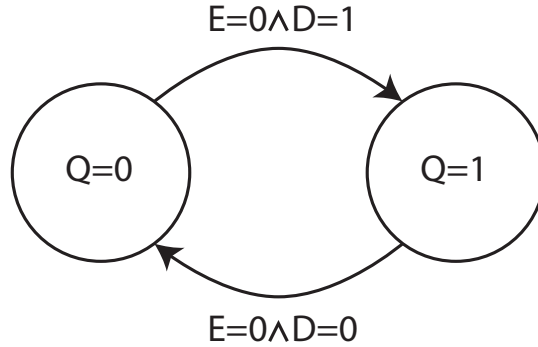


Figure 8.4: The state transition graph of a D-latch. This is an asynchronous graph, i.e. there is no implicit condition of a rising CLK signal for state transitions and the entire condition is explicitly stated with the state transition arrows. Note that any condition that is not listed along an arrow leading from a state will not change that state.

8.2 Synchronous flip-flops

Most digital electronics is **synchronous**. Synchronous means that there is a master digital signal, the so called clock (CLK), that defines the timing of state changes. Unlike a 'gating' signal it does not toggle between a phase where signals are latched and another phase where storage cells become transparent, i.e. between a phase where storage elements act as storage elements and a phase where they behave like combinational logic. Instead a clock signal makes synchronous storage elements latch the state of an input signal in a singular moment in time, i.e. when there is a *transition* of the clock signal (e.g. from 0 to 1, or vice versa, or even at both transitions). The input to synchronous storage elements may then change freely while the CLK is high or low without affecting the storage cell's content. Only at the exact moment of the appropriate *transition* will the storage cell take a 'snapshot' of the input signal and store it.

The most common synchronous storage cell is the D-flipflop. Its implementation with two D-latches is depicted in figure 8.3. It is often referred to as master-slave configuration of D-latches, where the first D-latch is the 'master'. While the CLK signal is HIGH the master keeps the input D as it was as CLK went HIGH and the slave is transparent. When the CLK goes LOW again the slave now latches the state that had been stored in the master, i.e. no change occurs at the output Q despite the master now being transparent. Only at the next transition of CLK from LOW to HIGH will the new state of the input D occur at the output Q . So the effect is that of a storage element that changes state only and exactly at the time of a LOW to HIGH transition, i.e. the rising flank of the CLK signal.

8.3 Finite state machines

Besides a circuit schematic or a characteristic table there is another way to describe the behaviour of a sequential logic circuit, i.e. a **state transition**

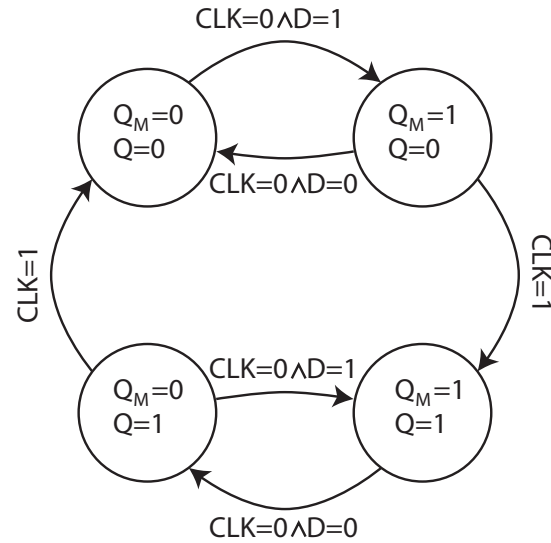


Figure 8.5: The state transition graph of a D-flipflop in asynchronous form. The D-flipflop is the basic synchronous storage element but here we give the graph that derives it from asynchronous D-latches. Seen as a whole with only its output Q as binary state variable, it can be described with a synchronous version of a state transition graph in figure 8.6. This asynchronous version however, makes the potential race condition between the input D and the clock signal CLK more visible: The timing of CLK and D if they change very close to each other in time is critical. If they change simultaneously, e.g. when $Q_M = 0, Q = 1$, it is not defined what should happen. It could be both a transition to $Q_M = 1, Q = 1$ or to $Q_M = 0, Q = 0$. To avoid this conundrum, synchronous circuits are carefully designed such that inputs are certain to be stable when the trigger flank of the clock occurs. (Such **race conditions** are generally a problem in asynchronous logic design if the change of two different signals lead away from a state to two different states.)

graph. This is a graph consisting of blobs and arrows between those blobs. The blobs are labelled with state identifiers and the arrows with conditions for state transitions. Thus the state transition diagram for the D-latch looks like depicted in figure 8.4.

The states in such a graph could theoretically be defined as the combination of all digital signal nodes in the circuit. In that manner one could even describe purely combinational logic with state-transition graphs too. However, usually one uses only the states of digital signal nodes that are able to maintain their state 'autonomously', i.e. their state is not fully defined by the the nodes defined as inputs to the circuit, but dependent on signal history. Another way of defining them is to say that they are the nodes that are needed in *addition* to the inputs to fully describe the state of the circuit after some settling time. So in short: the state of storage elements in the circuit.

The asynchronous state transition graph of the D-flipflop is given in figure 8.5. The state is defined by the two storage elements, the D-latches. The graph

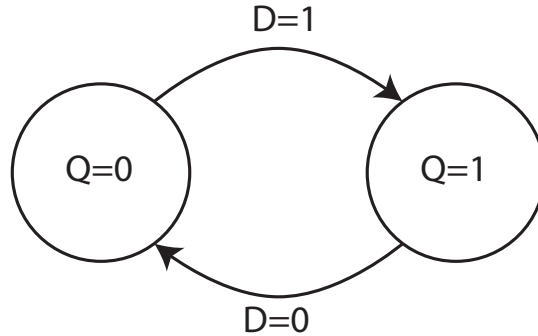


Figure 8.6: Synchronous D-flipflop state transition graph. This is the most usual form of state transition graphs, where the *states* are entirely defined by the content of synchronous storage elements. All state transitions have the implicit additional condition that they only occur at a specific transition of a clock signal.

becomes simpler as a synchronous state transition graph (figure 8.6). Only looking at what happens at the rising flank of the clock signal, Q and Q_M are interdependent and it's sufficient to only consider Q as the content of the D-flipflop.

The value of state transition graphs is not just as a way to describe a sequential logic circuit, but also as a means to describe an abstract function and to construct a sequential logic circuit with that specific function. Consider the example of a simplistic traffic light controller giving rise to the state transition graph on figure 8.7. This traffic light has but two states: either it is green for the NW direction and red for EW, or vice versa. It maintains its state unless one or more cars are waiting in the blocked direction. The implicit CLK signal is very slow in electronics terms, let's say the clock cycle lasts for a whole 20 seconds, so the minimum duration for the light to switch from one state to the other is 20 seconds.

There is a systematic way of deriving a logic circuit of the structure of a **finite state machine (FSM)** (see figure 8.8) from such a state transition graph.

First one assigns a different number to each of the states. In our example we can choose to assign '0' to 'green EW' and '1' to 'green NS', for example. Then one encodes these numbers as binary codes. Here, since we only have two states we can use a single bit and call that state variable 'GEW'. If there are more states one needs to use an appropriate number of 1-bit storage cells/variables. Binary numbers are the most straightforward way of encoding the states, but other encoding schemes might be chosen as well and might result in a different, and possibly simpler FSM. Anyhow, any encoding scheme works that assigns a unique bit-pattern to each state.

Then one also needs to encode the (sensor) input as binary signal. Here we have two sensors the state of which we encode as two Boolean variables 'EW' and 'NS'. The variables are '0' if no car is waiting and '1' if a waiting car is detected. Thus we can create a characteristic table and then a characteristic

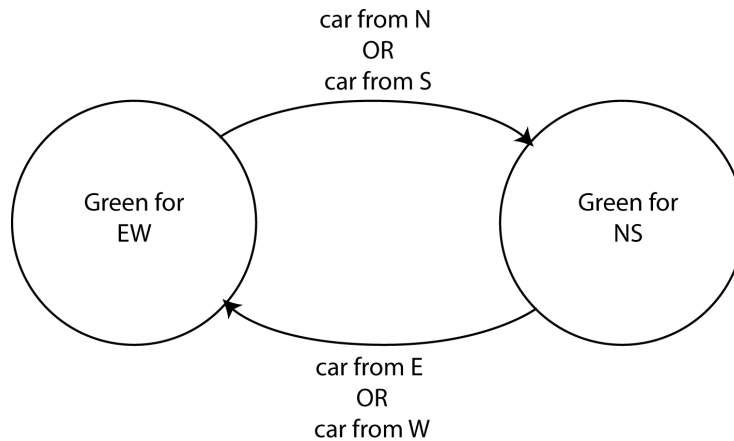


Figure 8.7: State transition graph of a traffic light controller that only changes state when a car is detected waiting in the blocked direction.

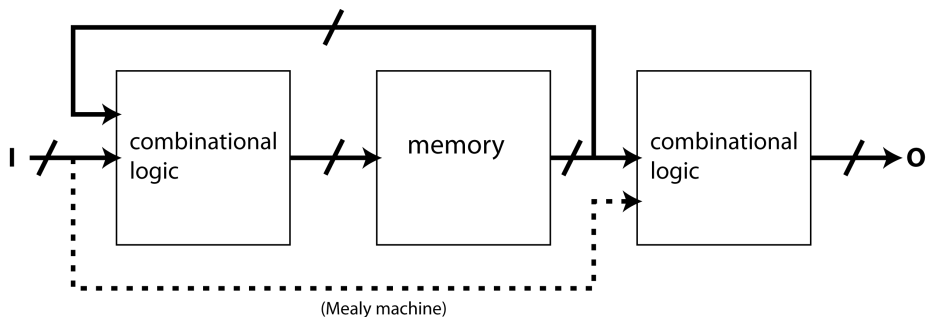


Figure 8.8: General structure of a **finite state machine (FSM)** that allows a direct realization from a state transition graph. The state is stored in the memory which consists of the necessary number of bit storage cells. The feed-back combinational logic implements the state transition conditions. The combinational logic to the output generates control signals from the state in the memory. This is known as a **Moore FSM**. A variant is the **Mealy FSM** that allows inputs to be used directly to generate output control signals (dashed arrow). This may allow to use fewer states but can be problematic if the inputs are not synchronized to the same CLK as the memory of the FSM. In that case the outputs may not be synchronized to that CLK either.

EW	NS	GEW	GEW
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Table 8.3: Characteristic table traffic light controller

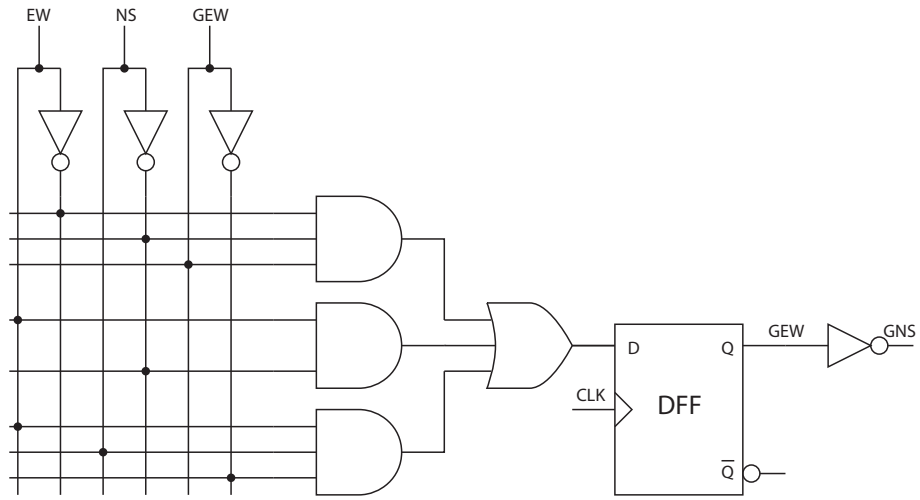


Figure 8.9: Schematic of the traffic light final state machine

function, i.e. list all possible combinations of internal state and input together with the resulting next state. The characteristic table is shown in table 8.3. From it one can derive a characteristic function for each state bit by for example use all rows where that bit turns '1', use a and gate for each row 'and-ing' the input states and internal states of each of those rows and 'or-ing' them together:

$$\begin{aligned}
 GEW = & \\
 & (\overline{EW} \wedge \overline{NS} \wedge GEW) \\
 & \vee (EW \wedge \overline{NS} \wedge \overline{GEW}) \\
 & \vee (EW \wedge \overline{NS} \wedge GEW) \\
 & \vee (EW \wedge NS \wedge \overline{GEW})
 \end{aligned} \tag{8.1}$$

This expression can then be used to make the combinational logic feedback circuit in the FSM and will get a functionally correct FSM. Optionally one can first try to simplify that expression somewhat. Here for example the middle two lines $(EW \wedge \overline{NS} \wedge \overline{GEW})$ and $(EW \wedge \overline{NS} \wedge GEW)$ can be reduced to a single

one $(EW \wedge \overline{NS})$:

$$\begin{aligned}
 GEW = & \\
 & (\overline{EW} \wedge \overline{NS} \wedge GEW) \\
 & \vee (EW \wedge \overline{NS}) \\
 & \vee (EW \wedge NS \wedge \overline{GEW})
 \end{aligned} \tag{8.2}$$

This results in the schematic in figure 8.9. Equation (8.2) is implemented as the combinational logic deriving input D to the one bit storage cell, the D-flipflop. Just for illustration a single inverter is added at the output as some output combinational logic generating the control signal that also turns the light in the NS direction green. This signal can of course also be taken directly from the $\overline{Q}$ port from the D-flipflop.

8.4 Example list of more advanced sequential logic circuits

For now and for this compendium, we will name some more sequential logic circuits only as a list. It will be outside the scope of the course to go into the design specifics of these. These are some examples of electronics that can still be usefully described as FSMs. For even more complex sequential electronics, one can for example use hierarchical FSMs that interact with eachother, i.e. it's not very useful to describe an entire CPU or Microcontroller as FSM, but parts of it can be, e.g. the arithmetic logic unit (ALU).

- counters
- appliance controllers (dishwashers, fridges, thermostats, ...)
- control circuit for SAR ADCs
- ALU (if it implements operations requiring multiple clock cycles)

Part IV

Analog Circuit Basics

Chapter 9

Single Ended Amplifiers

9.1 General transconductance amplifier low frequency/DC analysis

Figure 9.1 shows a version of a general low frequency small signal representation of an amplifier. This particular model is convenient if R_O is large and one can think of it as a transconductance amplifier. Normally it is considered connected to a signal source represented by a voltage source and its output resistance R_S and a load represented by load resistance R_L . When deriving the equivalent parameters R_I , G_M , and R_O from a concrete circuit model, R_S and R_L may influence the result, e.g R_S may influence the derivation of R_O , etc.

So for any single ended amplifier circuit, one can find a linearized version of it that complies with the structure in Fig. 9.1. We define and find R_I , G_M , and R_O at a particular biasing point/point of operation as:

$$R_I := \frac{v_i}{i_i} = \frac{\partial V_I}{\partial I_I} \quad (9.1)$$

$$G_M := \frac{i_o}{v_i} = \frac{\partial I_O}{\partial V_I} \text{ for } V_O \text{ held constant and } v_o = 0 ! \quad (9.2)$$

$$R_O := \frac{v_o}{i_o} = \frac{\partial V_O}{\partial I_O} \text{ for } V_{SIG} \text{ held constant and } v_{sig} = 0 ! \quad (9.3)$$

These definitions indicate two ways of finding these parameters. For both, you first have to find the point of operation in the large signal model, i.e. the bias currents and voltages V_I , I_I , V_O , and I_O . Then you either:

1. continue in the large signal domain and find the derivatives by applying small test signals around the point of operation.
2. replace the circuit elements with their small signal equivalents and find the small signal ratios by applying test signals.

So for example to find $\frac{\partial I_O}{\partial V_I}$ in the large signal domain you connect an ideal voltage source to the output that holds it constant at its point of operation V_O , and measure or compute ∂I_O when applying an infinitesimally small ∂V_I to the

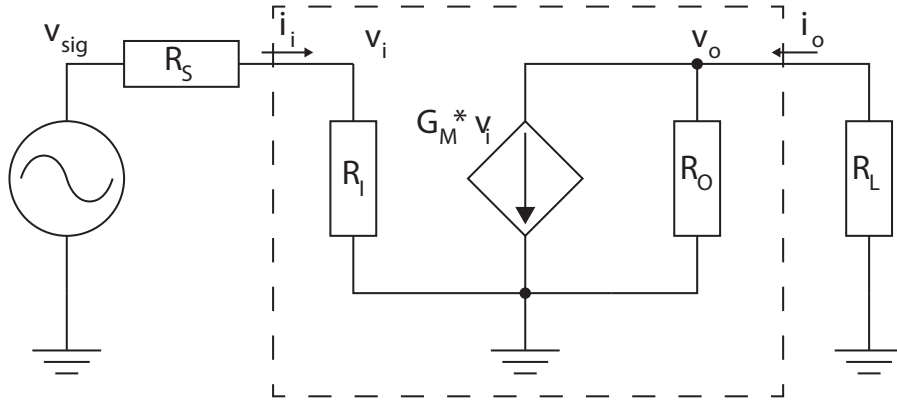


Figure 9.1: One possibility for a general linear (or small signal) model of an amplifier is depicted inside the dashed box. This particular model is convenient if R_O is large and one can think of it as a transconductance amplifier. Normally it is considered connected to a signal source represented by a voltage source and its output resistance R_S and a load represented by load resistance R_L . Both R_S and R_L can have an influence when deducing the amplifier parameters R_O and R_I respectively.

input. Or you replace all circuit elements with their small signal equivalents, connect v_o to Gnd, and compute $\frac{v_o}{i_o}$.

We will refer to the maximal voltage gain of the amplifier, i.e with the ideal (but usually unachievable) configuration where $R_S = 0$ and $R_L = \infty$, as **intrinsic gain** A_O :

$$A_O = -G_M R_O \quad (9.4)$$

Be aware that we defined $G_M := \frac{i_o}{v_i}$ and that if you want to know what $\frac{i_o}{v_{sig}}$ is, you have to take into consideration that there is a resistive voltage division happening between R_S and R_I , i.e. $v_i = v_{sig} \frac{R_I}{R_S + R_I}$. Consequently we have:

$$\frac{i_o}{v_{sig}} := G'_M = G_M \frac{R_I}{R_S + R_I} \quad (9.5)$$

Also be aware that the voltage gain $\frac{v_o}{v_i}$ of this amplifier is severely influenced by R_L . So if you design the circuit to have a large R_O in order to achieve a large voltage gain, then you have to pay close attention that R_L is of the same order as R_O or ideally even bigger, because:

$$\frac{v_o}{v_i} = -G_M (R_O || R_L) \quad (9.6)$$

So if you want to know the voltage gain A' starting from v_{sig} taking both R_S and R_L into account then it becomes:

$$A' := \frac{v_o}{v_{sig}} = -G'_M (R_O || R_L) = \frac{R_I}{R_S + R_I} A_O \frac{R_L}{R_O + R_L} \quad (9.7)$$

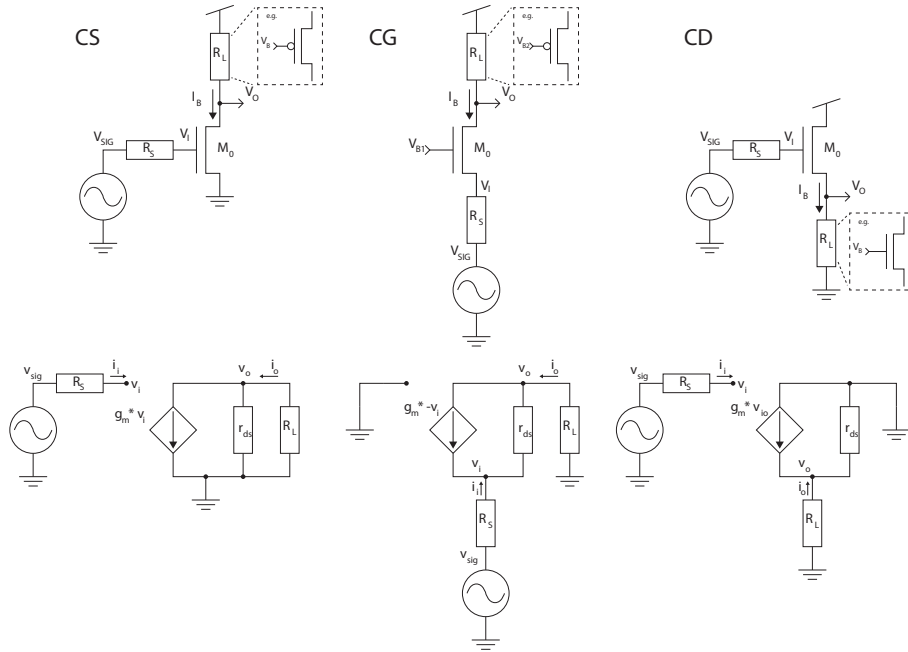


Figure 9.2: The common source (CS), common gate (CG) and common drain single FET amplifier configurations. Large signal representation on top and the corresponding small signal representations below. The load resistance R_L in a CMOS integrated circuit is usually implemented with another transistor, such that $R_L = r_{ds}$ in the small signal representation.

9.2 Single transistor amplifier low frequency/DC analysis

In the following we will introduce three single transistor gain stages, simply by using different pairs among the transistor pins as input and output. Then by deriving the small signal R_I , G_M , and R_O under specific biasing conditions (summarized in table 9.1) we can derive the gain of all these stages from the general transconductance amplifier model in figure 9.1 and transfer function (9.7).

9.2.1 Biasing/point of operation

Note that all transistors used in these stages need to be biased (i.e. the point of operation (POO) needs to be chosen) in the saturation region to work as discussed in the following. Thus, the small signal r_{ds} is considerable (the inverse of the slope of the usually very flat slope of the I_D vs. V_{GS} curve in saturation) and the term $g_m r_{ds} \gg 1$ is big and thus $\frac{1}{g_m} \ll r_{ds}$ and $\frac{1}{r_{ds}} \ll g_m$. Consequently, a condition for a useful biasing point in the large signal domain for these circuits is that $V_{DS} > V_{sat}$! This often leads to some restrictions, not just for the POO currents and voltages, but also for the input and output ranges, e.g. that an amplifier will not deliver an output swing all the way from Gnd/Vss to Vdd.

9.2.2 Common Source (CS) gain stage

Now you might actually realize that this general transconductance amplifier model (figure 9.1) resembles our small signal transistor model considerably. Most obviously the **Common Source (CS)** amplifier we have already introduced as a small signal analysis example in figure 3.6 chapter 3 fits very nicely into this scheme. We simply declare $v_i := v_g$, $v_o := v_d$, and $v_s := 0$ is Gnd and voilà: we have ourselves an amplifier just like the one in figure 9.1. The name 'common source' actually comes from the fact that the source terminal is the common reference for both input and output. It is depicted in figure 9.2, in the left column labelled CS. The amplifier parameters are listed in table 9.1. In CMOS technology the initial load is usually provided by another FET transistor with a fixed $V_B = V_{GS}$. If the output is further connected to another circuit's input, that circuit's input resistance would also need to be added in parallel for the total R_L . If however, it is not connected to anything else, the r_{ds} of that **biasing transistor** is all of R_L . The bias current is then the current provided by that transistor and needs to be matched approximately by the CS transistor (setting an appropriate DC-level for the input signal V_{SIG}) to obtain an appropriate point of operation. If the currents are not matched, the output will simply saturate towards either Gnd or Vdd, and the circuit cannot operate as intended.

The CS amplifier achieves good voltage gain because of the high output resistance together with the good transconductance. However, it cannot drive loads with small resistances without loosing that gain. In the case of a very small resistance as load it is better to think of it as a transconductance amplifier, i.e. where it delivers a reliable current $i_o \approx g_m v_i$ to the load.

	R_I	G_M	R_O
CS	∞	g_m	r_{ds}
CG	$\frac{r_{ds} + R_L}{1 + g_m r_{ds}}$	$-\left(\frac{1}{r_{ds}} + g_m\right)$	$r_{ds} + R_S(g_m r_{ds} + 1)$
CD	∞	$-g_m$	$\frac{1}{g_m} r_{ds} = \frac{r_{ds}}{1 + g_m r_{ds}}$
Cascode	∞	$g_{m1} r_{ds1} \frac{\frac{r_{ds2}}{1 + g_{m2} r_{ds2}}}{r_{ds1} + \frac{r_{ds2}}{1 + g_{m2} r_{ds2}}} \left(\frac{1}{r_{ds2}} + g_{m2}\right)$	$r_{ds2} + r_{ds1}(g_{m2} r_{ds2} + 1)$

Table 9.1: Equivalent linear amplifier parameters for single transistor amplifier- and the cascode configurations. For a deduction from Kirchhoff current equations for the most complicated case, the CG, have a look at the appendix, section B.2.1

9.2.3 CS/transistor intrinsic gain

Note that the previously defined **intrinsic gain** for a general transconductance amplifier often is used to more specifically refers to the intrinsic gain of a CS connected single transistor, i.e. the best gain A CS stage FET can possibly achieve (i.e. with an infinite load resistance) at a given biasing point. Thus, using the values for the CS stage in table 9.1, the intrinsic gain for a single transistor is:

$$A_0 = g_m r_{ds} \quad (9.8)$$

So how do we design a CS stage for maximum intrinsic gain? What are the parameters we can tweak? If we assume that we are stuck with a specific CMOS process and thus the given process parameters, we can still tune the design parameters, i.e. W and L , as well as the biasing point (i.e. the point of operation, nominally V_{OV}). Let's look at these in turn. We assume to be in saturation as r_{ds} becomes instantly much much smaller when entering conduction and one normally never considers conduction as a useful region of operation for a CS stage.

In strong inversion we get (note that the product λL is constant, i.e. a process parameter)

$$\begin{aligned}
g_m &= \mu C_{ox} \frac{W}{L} V_{OV} (1 + V_{DS} \lambda) \approx \mu C_{ox} \frac{W}{L} V_{OV} \\
r_{ds} &= \frac{1}{\lambda I_D'} = \frac{2}{\lambda \mu C_{ox} \frac{W}{L} V_{OV}^2} \\
A_0 &= \frac{2L}{\lambda L V_{OV}}
\end{aligned} \quad (9.9)$$

In weak inversion we get

$$\begin{aligned}
g_m &= \frac{I_D}{U_T} = 2nkU_T e^{\frac{V_G - V_{tn} - nV_S}{nU_T}} \\
r_{ds} &= \frac{1}{\lambda I_D} = \frac{1}{\lambda 2nkU_T^2 e^{\frac{V_G - V_{tn} - nV_S}{nU_T}}} \\
A_0 &= \frac{L}{\lambda L U_T}
\end{aligned} \quad (9.10)$$

Thus, an increase in L increases A_0 proportionally, and in *strong inversion* an increase in V_{OV} decreases A_0 proportionally, while in *weak inversion* the gain is independent of V_{OV} .

9.2.4 The Common Gate (CG) gain stage

Now, a FET is a three terminal device¹ and can be hooked up differently. In fact, there are two more configurations that also work as amplifiers, and you might already suspect what they shall be called. First, the **Common Gate (CG)** gain stage/amplifier where $v_i := v_s$ and $v_o := v_d$, and $v_g := 0$ is connected to Gnd.

The CG amplifier with an ideal voltage source as signal behaves much the same as the CS amplifier without inverting the output, i.e. a decent voltage amplifier. However, due to its low input resistance it is heavily affected by signal source output resistance R_S (Compare table 9.1). It is better suited and thought of as a **current conveyor**: if one uses a current source i_{sig} with R_S in parallel rather than a voltage source v_{sig} , and if $R_S \gg \frac{1}{g_m}$, then since $R_I \approx \frac{1}{g_m}$ (as long as $R_L \leq r_{ds}$), the input voltage is $v_i \approx \frac{i_{sig}}{g_m}$. And since the transconductance is approximately $G_M \approx -g_m$, the output current will be approximately $i_o \approx i_{sig}$ and this current will be little affected by a resistive load (as long as it is smaller than the considerable output resistance). This is thus useful to convey a current from a source with not so high output resistance R_S to an output with very high output resistance $R_O = r_{ds}$.

9.2.5 The Common Drain (CD) stage or Source Follower (SF)

And finally the **Common Drain (CD)** stage, better known as **Source Follower (SF)**, where $v_i := v_g$, $v_o := v_s$ and $v_d := 0$ is the common Gnd reference. The equivalent transconductance amplifier parameters G_M, R_I, R_O can be deduced from Kirchhoff current equations and are listed in table 9.1.

The CD/SF stage is not really an amplifier, since its voltage gain is close to 1. As the name 'source follower' suggests, it is used as a follower. In the small signal world the output is simply a copy of the input, with a small gain error: the intrinsic gain is $A_0 = -G_M R_O = \frac{g_m r_{ds}}{1 + g_m r_{ds}}$, i.e. the gain if $R_L \gg \frac{1}{g_m} || r_{ds}$. In the large signal world it follows its input with an offset, i.e. the gate to source voltage V_{GS} that is determined by the biasing current in the branch. In contrast to the CS stage and dependent on the large signal biasing current's magnitude it is quite capable of driving resistive loads, i.e. it has a small output resistance R_O . It may thus be used as a second stage after a CS stage to maintain the gain, while making the output less sensitive to the load. So it can turn a not so good voltage source with a relatively large R_S into a better voltage source with its smaller R_O .

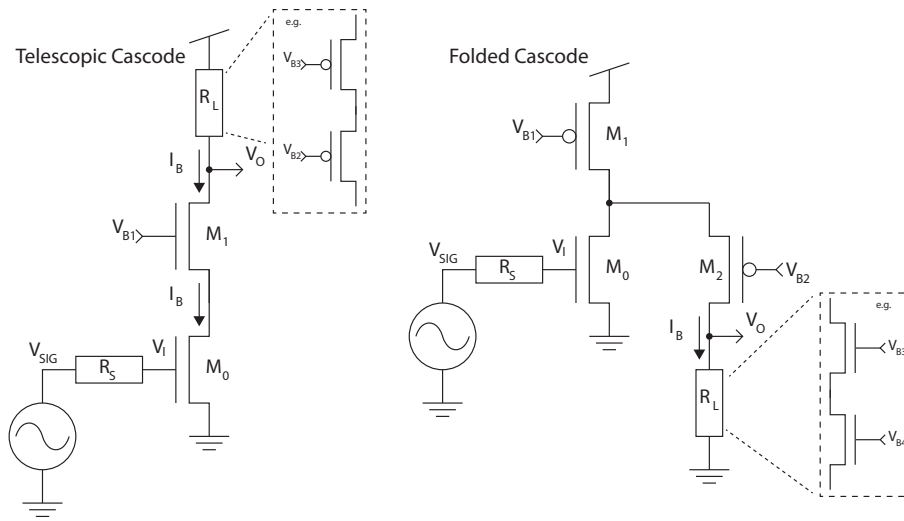


Figure 9.3: The telescopic and folded cascode gain stage. The telescopic variant modulates the current into a CG stage directly from a CS stage. The telescopic variant does the same by subtracting the CS output current from the input current to the CG stage M_2 of the opposite transistor type. Note that the saturation current of bias transistor M_1 needs to supply both the saturation bias current through M_0 and R_L (i.e. V_{B4}) to place the point of operation for V_O close to $\frac{V_{dd}}{2}$. The small signal representations of the telescopic and the folded Cascode gain stages are equivalent barring an extra parallel r_{ds} loading the CS drain.

9.3 The Cascode gain stage

A **Cascode** gain stage depicted in figure 9.3 is a CS and CG in series. The total circuit's output resistance is that of the CG stage, with the output resistance of the CS stage as the CG's signal source resistance. The total input resistance is infinite. The total transconductance is that of the CG stage multiplied with the voltage gain of the CS stage with the input resistance of the CG stage as load when $R_L = 0$. Using index 1 for the CS transistor and index 2 for the CG transistor:

$$R_O = r_{ds2} + R_{O1}(g_{m2}r_{ds2} + 1) = r_{ds2} + r_{ds1}(g_{m2}r_{ds2} + 1) \quad (9.11)$$

$$G_M = -A'_1|_{(R_L=0)} \left(\frac{1}{r_{ds}} + g_m \right) \quad (9.12)$$

$$= g_{m1}r_{ds1} \frac{R_{I2}|_{(R_L=0)}}{r_{ds1} + R_{I2}|_{(R_L=0)}} \left(\frac{1}{r_{ds2}} + g_{m2} \right) \quad (9.13)$$

$$= g_{m1}r_{ds1} \frac{\frac{r_{ds2}}{1+g_{m2}r_{ds2}}}{r_{ds1} + \frac{r_{ds2}}{1+g_{m2}r_{ds2}}} \left(\frac{1}{r_{ds2}} + g_{m2} \right) \quad (9.14)$$

$$R_I = \infty \quad (9.15)$$

If one makes the assumptions that in general $g_m r_{ds} \gg 1$ and $g_{m1} \approx g_{m2}$ and $r_{ds1} \approx r_{ds2}$ one can simplify these set of equations to:

$$R_O \approx g_m r_{ds}^2 \quad (9.16)$$

$$G_M \approx g_m \quad (9.17)$$

$$R_I = \infty \quad (9.18)$$

So note that the actual gain with a load R_L , i.e. $A' = -G_M(R_O || R_L)$, will depend a lot on that R_L . If one uses just a single pFET on top of a nFET cascode stage, $R_L = r_{dsp}$. If the drain resistance of a pFET is similar to that of the nFETs (i.e. $r_{dsp} = r_{dsn} = r_{ds}$) then:

$$A' = -G_M(R_O || R_L) = g_m(g_m r_{ds}^2 || r_{ds}) \approx g_m r_{ds} \quad (9.19)$$

And A' is practically no bigger than that of a CS stage. Interestingly though, such a configuration actually can increase the *bandwidth* as compared to just the CS stage, as we will see in subsection 9.7. If however, we really want more gain, the solution is to use *two* biasing transistors, i.e. by mirroring the cascode structure (the 'insets' in figure 9.4). The output resistance of two biasing transistors in series can be derived exactly like the one of the cascode gain stage. So now we have:

$$R_L = g_m r_{ds}^2 \quad (9.20)$$

And consequently:

$$A' = -G_M(R_O || R_L) = g_m(g_m r_{ds}^2 || g_m r_{ds}^2) = \frac{1}{2} g_m^2 r_{ds}^2 \quad (9.21)$$

¹assuming V_B and V_S are shorted and/or that V_B is fixed at Gnd.

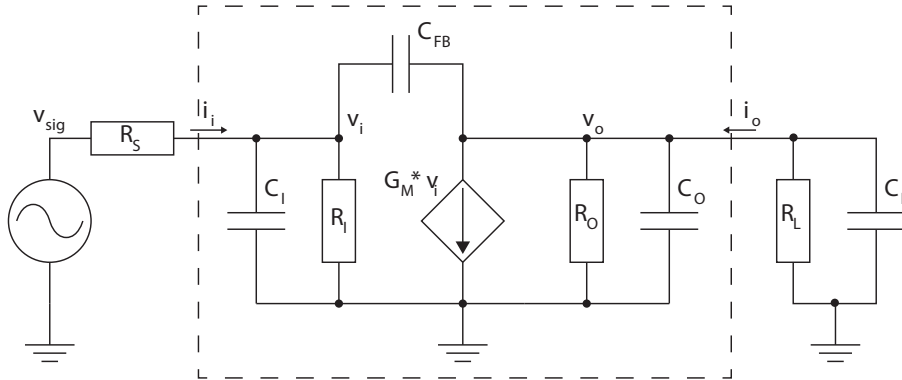


Figure 9.4: General transconductance amplifier model for frequency analysis

General transamp	C_I	C_{FB}	C_O
CS	C_{GS}	C_{GD}	C_{DB}
CG	$C_{GS} + C_{SB}$	N/A	$C_{GD} + C_{SD}$
SF/CD	C_{GD}	C_{GS}	C_{SB}

Table 9.2: Single transistor gain stages capacitors assigned to the general transconductance amplifier model.

9.4 General transconductance amplifier frequency response

So far we have neglected the 'parasitic' capacitors in the small signal model. Important will be C_{GS} , C_{GD} , C_{DB} , and C_{SB} . Mostly we will consider the bulk being connected to the source, so often-times we can ignore C_{SB} . In general our three terminal amplifier model (fig 9.1) will gain an input capacitor and an output capacitor to Gnd, as well as a capacitor between input and output, resulting in figure 9.4. For the three single transistor gain stages these three capacitors correspond to the transistor model capacitors as indicated in table 9.2.

9.4.1 Exact solution

So let us look at the general transconductance amplifier model of figure 9.4 (redrawn with some colour-coding in figures 9.5 and 9.6) that is appropriate for single stage single ended transconductance amplifiers, and figure out its transfer function. We can then apply this general solution to our single transistor gain stages later on.

Let us apply the systematic method as introduced in 5.2 to set up the Kirchhoff current law equations for the two nodes v_i and v_o ². v_{sig} is a source and comes from an ideal voltage source, so it's not an unknown and can thus be

²We do this in great detail here as it is ment to serve as an example of applying the method introduced in section 5.2.

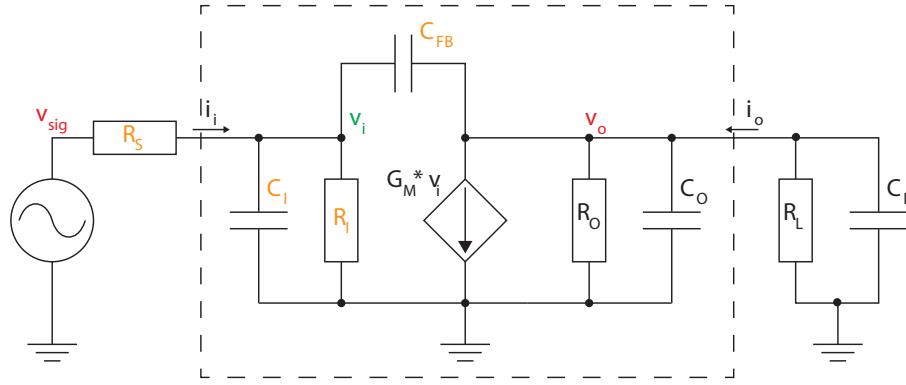


Figure 9.5: Color marked general transconductance amplifier illustrating derivation of Kirchhoff's current law equation for node v_i .

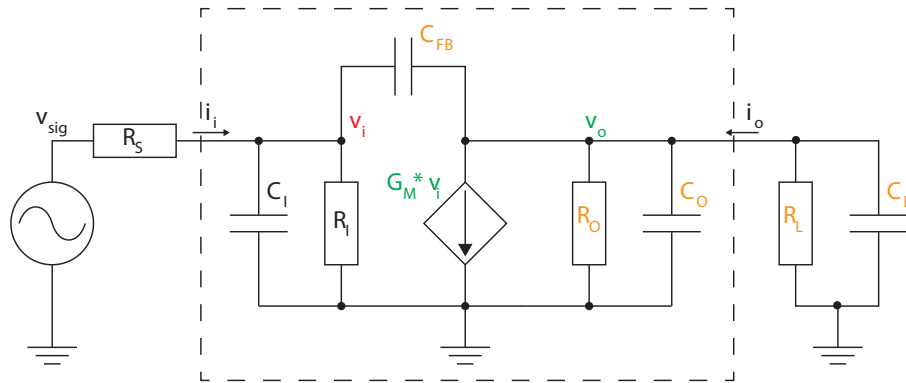


Figure 9.6: Color marked general transconductance amplifier illustrating derivation of Kirchhoff's current law equation for node v_o .

ignored here. So for node v_i you may look at figure 9.5 where some labels are marked in colour for easier understanding. To simplify the following expressions a bit we combine the parallel resistors into $R_L^* := R_O || R_L$ and parallel capacitors $C_L^* := C_L || C_O$. Also useful will be an expression $R_I^* = R_I || R_S$. We construct the equation by placing all contributions to current flowing *out* of v_i on the left hand side and all contributions to currents flowing *into* v_i on the right hand side: we multiply v_i (marked in green) with all admittances (i.e. impedances need to be inverted!) connected to it (marked in orange) and place that term on the left hand side. Furthermore we would add any explicit current sources of currents entering v_i on the left hand side as well (see the derivation of equation for v_o .) For the right hand side we multiply the same conductances with the voltages on their other side (marked in red) and we would add any explicit current sources of currents *into* v_i . Note that the node Gnd is defined as zero and thus is ignored. This yields equation (9.22). Doing the same for node v_o (colour marked figure 9.6) yields equation (9.23). Note the explicit current source added to the left hand side, as it is a current flowing *out* of v_o .

$$v_i \left(\frac{1}{R_S} + \frac{1}{R_I} + s(C_{FB} + C_I) \right) = v_{sig} \frac{1}{R_S} + v_o s C_{FB} \quad (9.22)$$

$$v_o \left(\frac{1}{R_L^*} + s(C_{FB} + C_L^*) \right) + v_i G_M = v_i s C_{FB} \quad (9.23)$$

Solving this to get the transfer function $\frac{v_o}{v_{sig}}$ yields a rather complicated expression:

$$\begin{aligned} \frac{v_o}{v_{sig}} &= -G_M R_L^* \frac{R_I}{R_I + R_S} \frac{(1 - s \frac{C_{FB}}{G_M})}{1 + sB + s^2 A} \\ B &= R_L^* (C_{FB} + C_L^*) + R_I^* (C_I + C_{FB}(1 + G_M R_L^*)) \\ A &= R_I^* R_L^* [(C_{FB} + C_L^*)(C_{FB} + C_I) - C_{FB}^2] \end{aligned} \quad (9.24)$$

This general result is OK to make a Bode plot with a maths program on a computer but way too complicated to easily read any directly useful information out of it. However, making some assumptions of extreme cases we can glean some useful practical information.

9.4.2 Without feedback capacitor

A simple special case, for instance, is when there is no feedback at all $C_{FB} = 0$ and the expression becomes:

$$\frac{v_o}{v_{sig}} = -G_M R_L^* \frac{R_I}{R_I + R_S} \frac{1}{(1 + sR_I^* C_I)(1 + sR_L^* C_L^*)} \quad (9.25)$$

This is then simply two 1^{st} order filters in series with poles at frequencies $\frac{1}{R_I^* C_I}$ from the first stage and $\frac{1}{R_L^* C_L^*}$ from the second and two individual DC gains of $\frac{R_I}{R_I + R_S}$ from the first and $-G_M R_L^*$ from the second, and the total DC gain being the product of the two $-G_M R_L^* \frac{R_I}{R_I + R_S}$.

9.4.3 Assuming dominant pole at output

If the input node of the general transconductance amplifier is directly driven from the ideal voltage source and thus not affected by high frequency, i.e. if we consider $R_S \approx 0$, then $v_i \approx v_{sig}$ and only v_o will experience any influence of frequency and equation (9.24) becomes a simple first order low pass transfer function with one pole and one zero.

$$\frac{v_o}{v_{sig}} = -G_M R_L^* \frac{1 - s \frac{C_{FB}}{G_M}}{1 + s R_L^* (C_{FB} + C_L^*)} \quad (9.26)$$

So we get one pole and one zero:

$$\omega_{p_o} = \frac{1}{\tau_o} = \frac{1}{R_L^* (C_{FB} + C_L^*)} \quad (9.27)$$

$$\omega_{z_o} = \frac{G_M}{C_{FB}} \quad (9.28)$$

Note that the unity gain frequency ω_T for this transfer function (9.26) is (just using the first order a 20dB/dec roll off from ω_{p_o} and assuming the zero does not influence that roll off):

$$\omega_T = |A_{DC}| \omega_{p_o} = \frac{G_M}{C_{FB} + C_L^*} \quad (9.29)$$

and thus as long as $G_M > \frac{1}{R_L^*} \Leftrightarrow |A_{DC}| > 1$ we can state that:

$$\omega_{p_o} < \omega_T < \omega_{z_o} \quad (9.30)$$

So the zero after the unity gain frequency does indeed usually not have an important impact.

Note: More generally speaking the precise condition for a dominant pole at the output is that the resulting time constant from the assumption that there IS a dominant pole at the output $\tau_o := R_L^* (C_{FB} + C_L^*)$ needs to be much bigger than the time constant that we will derive from the assumption that there is a dominant pole at the input $\tau_i := R_I^* (C_I + C_{FB} (1 + G_M R_L^*))$ derived from the assumption that there is no frequency dependency at the output in the next section 9.4.4.

$$R_L^* (C_{FB} + C_L^*) \gg R_I^* (C_I + C_{FB} (1 + G_M R_L^*)) \quad (9.31)$$

If those two time constants are too close together (rule of thumb within a factor of 4) we cannot assume a dominant pole and the full equation (9.24) has to be used.

9.4.4 Assuming dominant pole at input

On the other hand if we consider the output to be simply $v_o = G_M R_L^* v_i$ and only v_i to be affected by frequency, we can modify (9.23) to

$$v_o \frac{1}{R_L^*} + v_i G_M = 0 \quad (9.32)$$

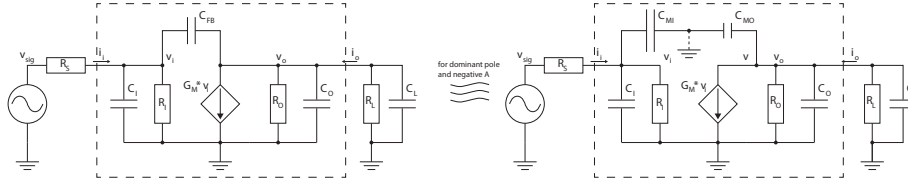


Figure 9.7: An illustration of the 'Miller effect' that makes the feedback capacitor C_{FB} appear much bigger at the input of an inverting amplifier. To derive a dominant pole at the input node, it can be replaced with two capacitors in series of the right size to create a virtual Gnd between them. This way they can be considered capacitors adding in parallel to the input and an output capacitors. If the dominant pole is at the input this can directly be used to derive this dominant time constant. (Caveat: This is safely true only as long as the DC relation $v_o = v_i G_M (R_O || R_L)$ is not impaired. So if that pole at node v_i is not clearly dominant, this approximation becomes imprecise.)

Note that this is a rather strong assumption! So this really only works if the pole resulting at the input node is dominant. If on the other hand v_o starts to decline at lower frequencies, then the negative feedback current flowing through C_{FB} to the input node will be much diminished and the pole we derive here will be pushed to higher frequencies in the real system, i.e. when using the full equation (9.24). So be aware of that limitation!

OK, but if we accept that the input node is affected by the dominant pole and not the output node directly, then (9.24) becomes:

$$\frac{v_o}{v_{sig}} = -G_M R_L^* \frac{R_I}{R_S + R_I} \frac{1}{1 + s R_I^* (C_I + C_{FB}(1 + G_M R_L^*))} \quad (9.33)$$

So we get

$$\begin{aligned} \tau_i &= \frac{1}{\omega_{p_i}} = R_I^* (C_I + C_{FB}(G_M R_L^* + 1)) \\ &= R_I^* (C_I + C_{FB}(A_{DC} + 1)) \end{aligned} \quad (9.34)$$

Again equivalently to the more precise condition for a dominant pole at the output, the inverse more general test applies for this dominant pole at the input:

$$R_L^* (C_{FB} + C_L^*) << R_I^* (C_I + C_{FB}(1 + G_M R_L^*)) \quad (9.35)$$

9.4.5 Miller approximation assuming dominant pole at input

Figure 9.7 offers a simplified view that yields the same result when assuming a dominant pole at the input, where the serial capacitor between input and output can be changed into equivalent capacitors at the input and output towards Gnd:

	$\tau_i = \frac{1}{\omega_{p_i}}$
CS	$R_S (C_{GS} + C_{GD}(g_m(r_{ds} R_L) + 1))$
CG	$\left(R_S \frac{r_{ds} + R_L}{1 + g_m r_{ds}}\right) (C_{GS} + C_{SB})$
CD	$R_S \left(C_{GD} + C_{GS} \left(1 - g_m \left(R_L \frac{1}{g_m} r_{ds}\right)\right)\right)$

Table 9.3: Single transistor amplifiers time constants when the input node dominates

	$\tau_o = \frac{1}{\omega_{p_o}}$	ω_{z_o}
CS	$(r_{ds} R_L)(C_{GD} + C_{DB} + C_L)$	$\frac{g_m}{C_{GD}}$
CG	$[(r_{ds} + R_S(g_m r_{ds} + 1)) R_L](C_{GD} + C_{DB} + C_L)$	NA
CD	$\left(R_L \frac{1}{g_m} r_{ds}\right) (C_{GS} + C_{SB} + C_L)$	$\frac{-g_m}{C_{GS}}$

Table 9.4: Single transistor amplifiers time constants when the output node dominates

$$C_{MI} = (1 + A_{DC})C_{FB} = (1 + G_M R_L^*)C_{FB} \quad (9.36)$$

$$C_{MO} = \left(\frac{1}{A_{DC}} + 1\right)C_{FB} \quad (9.37)$$

This is under the assumption that the DC gain is negative/inverting and the gain between v_i and v_o is not yet impaired by a pole or zero: so this simplification can be safely adopted to find a dominant pole at the node v_i of a low pass filter. With this simplification the input section and output section remain independent stages and one can derive a separate transfer function for the input stage, while the transfer function from v_i to v_o is equivalent to its DC transfer function $\frac{v_o}{v_i} = G_M R_L^*$. This RC configuration at the input directly leads to the same pole as in (9.34) and the resulting transfer function is exactly the same as (9.33).

Note that using the Miller approximation for a dominant pole at the output using C_{MO} does not work quite so well: the condition that $\frac{v_o}{v_i} = G_M R_L^*$ is not valid at the cut off frequency. Thus the analysis for a dominant pole at the output in equation (9.26) does NOT yield a pole that is $\frac{1}{R_L^* (C_O + C_{FB} (\frac{1}{A_{DC}} + 1))}$. Quite the contrary: from equation (9.26) the pole frequency is higher as C_{FB} appears without influence of $G_M R_L^*$ and the zero frequency $\frac{G_M}{C_{FB}}$ may move the cut off to even higher frequencies³.

9.5 Single transistor gain stages frequency response

The general transconductance amplifier transfer function (9.24) can be applied to the three single transistor gain stage cases using the appropriate parameters

³In digital circuits the feed forward Miller effect onto the output node *is* in fact used, albeit slightly differently. See 7.1.3!

as in tables 9.1 and 9.2. If we just care for the two cases where either the dominant pole is at the input node or at the output node, we can use the simpler expressions in equations (9.33) and (9.26), yielding tables 9.3 and 9.4.

One notes that the CS stage is the most susceptible for a dominant pole at the input node, due to the Miller effect. So a small signal source output resistance R_S is essential for a good bandwidth. With respect to the output node the CG stage is the one with the amplified R_O and thus a potentially large R_L^* (dependent on R_L) and a tendency for a dominant pole at its output.

9.5.1 Intrinsic CS/transistor speed

When it comes to maximum frequency that can be achieved by a CS stage under ideal circumstances there is (analogous to the maximum possible gain, the *intrinsic gain*) the so called **transition frequency** ω_T , or more descriptively also known as **unity (current) gain frequency**. For the current gain considered here, it is the frequency at which the output current from the drain (connected to small signal Gnd) of a CS configured FET is no bigger than the input current. So if one assumes that the load connected to the drain includes a capacitive load C_L^* that is of similar magnitude than the input capacitance C_I (i.e. C_{GS} for the CS stage), also the voltage gain will be smaller or equal to 1. So an effective upper limit to the unity gain bandwidth of such a CS stage (just as the intrinsic gain is the upper limit for the gain). The condition of unity current gain translates as:

$$\begin{aligned} |i_o| &= |i_i| \\ |v_i|(\omega_T(C_{GD} + C_{GS})) &= |i_i| \\ |i_o| &= |v_i(g_m - j\omega_T C_{GD})| \\ |i_o| &\approx |v_i|g_m \text{ assuming } g_m \gg C_{GD} \\ |v_i|\omega_T(C_{GD} + C_{GS}) &\approx |v_i|g_m \end{aligned} \quad (9.38)$$

Thus:

$$\omega_T \approx \frac{g_m}{C_{GS} + C_{GD}} \quad (9.39)$$

One gets the same result if deriving the *voltage* unity gain frequency with an ideal voltage source input assuming that $C_L^* = C_{GS}$ and realizing that $C_{FB} = C_{GD}$ using (9.29).

How can one optimize the intrinsic speed of a CS stage/FET by manipulating the design parameters W , L and the point of operation as given by V_{OV} ?

The capacitances in dependency of those parameters are:

$$\begin{aligned} C_{GS} &= \frac{2}{3}C_{OX}WL \\ C_{GD} &= C_{OX}WL_{OV} \end{aligned} \quad (9.40)$$

Thus, in strong inversion we get:

$$\begin{aligned} g_m &= \mu C_{OX} \frac{W}{L} V_{OV} \\ \omega_T &= \frac{\mu V_{OV}}{L(L_{OV} + \frac{2}{3}L)} \end{aligned} \quad (9.41)$$

In weak inversion we get (ignoring the slope factor, back gate effect, i.e. assuming $n = 1$):

$$g_m = \frac{I_D}{U_T} = 2\mu C_{OX} \frac{W}{L} U_T e^{\frac{V_{OV}}{U_T}}$$

$$\omega_T = \frac{2\mu U_T e^{\frac{V_{OV}}{U_T}}}{L (L_{OV} + \frac{2}{3}L)} \quad (9.42)$$

So increasing the length L reduces speed approximately quadratically (for $L \gg L_{OV}$), while increasing V_{OV} helps the speed, linearly for strong inversion and exponentially for weak inversion.

9.5.2 Trade off of intrinsic speed vs gain

Now remember the intrinsic gain in dependency of V_{OV} and L in equations (9.9) and (9.10). Increasing L helps the gain and hampers the speed in both strong and weak inversion, while V_{OV} helps the speed in both cases, but has a detrimental effect on the gain only in strong inversion.

A popular figure of merit for amplifiers is the so called **gain bandwidth product (GBW)**. Note that in a first order system the GBW is equivalent to the unity gain frequency, and we can get the BW by dividing the unity gain frequency by the DC gain.

So assuming a load $C_L^* = C_{GS}$, then ω_T here is the *voltage* unity gain frequency and we get the intrinsic BW dividing (9.39) by (9.8):

$$BW_0 = \frac{1}{r_{ds}(C_{GS} + C_{GD})} \quad (9.43)$$

So in dependency of V_{OV} and L we get in strong inversion:

$$BW_0 = \frac{\frac{1}{2}\mu V_{OV}^2 \lambda L}{L^3 (\frac{L_{OV}}{L} + \frac{2}{3})} \quad (9.44)$$

and in weak inversion:

$$BW_0 = \frac{2\mu U_T^2 e^{\frac{V_{OV}}{U_T}} \lambda L}{L^3 (\frac{L_{OV}}{L} + \frac{2}{3})} \quad (9.45)$$

9.6 Stability of single transistor stages

Note that the CS and CD/SF stages do in fact have a feedback capacitor from the output to the input node. So this is a possible source of instability which might be worthwhile to check. Note that we have derived the complete transfer function for the general transconductance amplifier model in equation (9.24). It is actually a second order LP function (If $R_I^* > 0$, so that normally means $R_S > 0$). Thus, combining that with the definition of the Q-factor in (5.27) one gets:

$$Q = \frac{\sqrt{A}}{B} = \frac{\sqrt{R_I^* R_L^* [(C_{FB} + C_L^*)(C_{FB} + C_I) - C_{FB}^2]}}{R_L^* (C_{FB} + C_L^*) + R_I^* (C_I + C_{FB}(1 + G_M R_L^*))} \quad (9.46)$$

CG stage

Just to confirm that we do not have a problem *without* the the feedback capacitor (among the three that is the **CG stage**), using the previous definitions for τ_i ((9.34) dominant pole at input) and τ_o ((9.27) dominant pole at output) for brevity and with $C_{FB} = 0$ they become $\tau_i = R_I^* C_I$ and $\tau_o = R_L^* C_L^*$, we are left with:

$$Q^2 = \frac{\tau_i \tau_o}{(\tau_i + \tau_o)^2} = \begin{cases} < \frac{1}{4} & \text{if } \tau_i \neq \tau_o \\ = \frac{1}{4} & \text{if } \tau_i = \tau_o \end{cases} \quad (9.47)$$

$$\Rightarrow Q \leq \frac{1}{2}$$

The above is true because it is generally true that

$$\frac{ab}{(a+b)^2} = \begin{cases} \frac{1}{4} & \text{if } a = b \\ < \frac{1}{4} & \text{if } a \neq b \end{cases} \quad (9.48)$$

for positive a and b (i.e. $a, b \in \mathbb{R}^+$).

So with $Q \leq \frac{1}{2}$ the stage is always stable with real poles, i.e. no resonance or even flat bandwidth extension.

CS stage

Including C_{FB} and using the full definitions (9.34) and (9.27) for $\tau_i = R_I^* (C_I + C_{FB}(G_M R_L^* + 1))$ and $\tau_o = R_L^* (C_{FB} + C_L^*)$ we get:

$$\begin{aligned} Q^2 &= \frac{R_I^* (C_I + C_{FB}) \tau_o - R_I^* R_L^* C_{FB}^2}{(\tau_o + \tau_i)^2} \\ &= \frac{(\tau_i - R_I^* C_{FB} G_M R_L^*)(\tau_o) - R_I^* R_L^* C_{FB}^2}{(\tau_o + \tau_i)^2} \\ &\leq \frac{(\tau_i - R_I^* C_{FB} G_M R_L^*)(\tau_o)}{(\tau_o + \tau_i)^2} \end{aligned} \quad (9.49)$$

If $G_M \geq 0$ on can further again conclude:

$$\begin{aligned} Q^2 &\leq \frac{\tau_i \tau_o}{(\tau_o + \tau_i)^2} \\ &\Rightarrow Q \leq \frac{1}{2} \end{aligned} \quad (9.50)$$

CD/SF stage

However, for the CD/SF stage $G_M = -g_m < 0$ and the 'Miller term' is close to zero, as $G_M R_L^* = -g_m (\frac{1}{g_m} ||r_{ds}|| R_L) \approx -1$. So if it were exactly zero ($1 + G_M R_L^* = 0$), we get again $\tau_i = R_I^* C_I$ and if we define $\tau_o' := \tau_o - R_L^* C_{FB} = R_L^* C_L^*$ we can rewrite (9.49) as (Note that we use both τ_o' AND τ_o):

$$\begin{aligned}
Q^2 &= \frac{(\tau_i + R_I^* C_{FB})(\tau_o' + R_L^* C_{FB}) - R_I^* R_L^* C_{FB}^2}{(\tau_o + \tau_i)^2} \\
&= \frac{\tau_i \tau_o' + \tau_i R_L^* C_{FB} + \tau_o' R_I^* C_{FB}}{(\tau_o + \tau_i)^2}
\end{aligned} \tag{9.51}$$

In this case Q is no longer limited and it can be shown that the condition for $Q^2 > \frac{1}{4}$ can be met if:

$$\begin{aligned}
\frac{1}{2} \left(\frac{R_I^* C_I}{R_L^* (C_L^* + C_{FB})} + \frac{R_L^* (C_L^* + C_{FB})}{R_I^* C_I} \right) - 1 &< \frac{2C_L^* C_{FB}}{C_I (C_L^* + C_{FB})} \\
\frac{1}{2} \left(\frac{\tau_i}{\tau_o} + \frac{\tau_o}{\tau_i} \right) - 1 &< \frac{2C_L^* C_{FB}}{C_I (C_L^* + C_{FB})}
\end{aligned} \tag{9.52}$$

This condition is certainly met when the left-hand side is at its minimum (which is zero), i.e. when:

$$\begin{aligned}
R_I^* C_I &= R_L^* (C_L^* + C_{FB}) \\
\tau_i &= \tau_o
\end{aligned} \tag{9.53}$$

because then the left hand term of (9.52) becomes zero and the right hand term is always bigger than zero.

So specifically for the CD/SF stage this is a *sufficient condition* derived here. More generally speaking the more similar the two terms of the equation ($R_I^* C_I \approx R_L^* (C_L^* + C_{FB})$) the more likely instability becomes. Using the specific parameters involved, this is:

$$R_S C_{GD} \approx (C_{GS} + C_{SB} + C_L) \left(\frac{1}{g_m} ||r_{ds}|| R_L \right) \tag{9.54}$$

Which you can get if:

$$R_S g_m \approx \frac{(C_{GS} + C_{SB} + C_L)}{C_{GD}} \tag{9.55}$$

That is that the ratio of the output resistance of the signal source and the output resistance of the CD/SF stage is similar to the ratio of the CD/SF input capacitance (excl. feedback cap) and the total capacitance at the CD/SF stage output (incl. feedback cap). So if one is so unfortunate to observe some resonance in a SF then one can try to reduce R_S or to increase C_L to get out of this funk.

Take home message

So the **take home message** here is that CS and CG by themselves do not give rise to resonance or instability, while a CD/SF stage can.

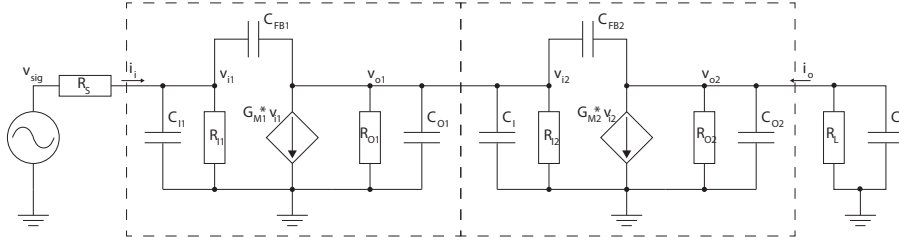


Figure 9.8: A cascode gain stage's frequency behaviour cannot be mapped on a single transconductance equivalent circuit as opposed to when just considering the low frequency behaviour. So one would need to consider all capacitances of both sequential single transistor stages as in this figure.

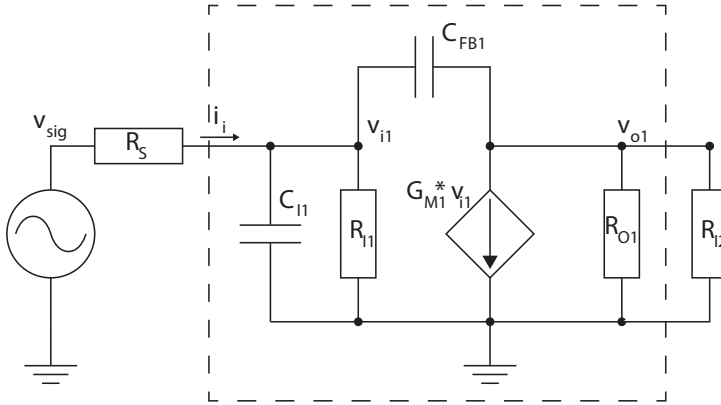


Figure 9.9: Cascode gain stage frequency model assuming dominant pole at input.

9.7 Cascode gain stage frequency behaviour

The frequency behaviour of the Cascode gain stage can no longer be fully described by the general transconductance amplifier model introduced here, because of an internal node and capacitors not represented in that model. Or put another way: the cascode gain stage is in essence a two stage circuit and the general transamp here is only appropriate for one stage circuits. One *could* thus employ the two equivalent models for the CS and CG stages in sequence like in figure 9.8, but that would already be quite complex and simulation is recommended rather than paper and pen analysis.

Still, if we assume a dominant pole on either the input node or the output node, we can draw a few conclusions. The cascode gain stage offers some interesting trade offs when tweaking R_L . Let's look at it again as a CS and a CG stage in series. Let us furthermore again simplify our ruminations to the two cases where a) the input node gives rise to the dominant pole, and b) the output node hosts a clearly dominant pole.

In case a) we can look at the CS stage with its load resistance being the input resistance $R_{I2} = \frac{r_{ds} + R_L}{1 + g_m r_{ds}}$ (see table 9.1) to the CG stage (Fig. 9.9). R_{I2} heavily depends on the R_L at the cascode output. We have learnt that we

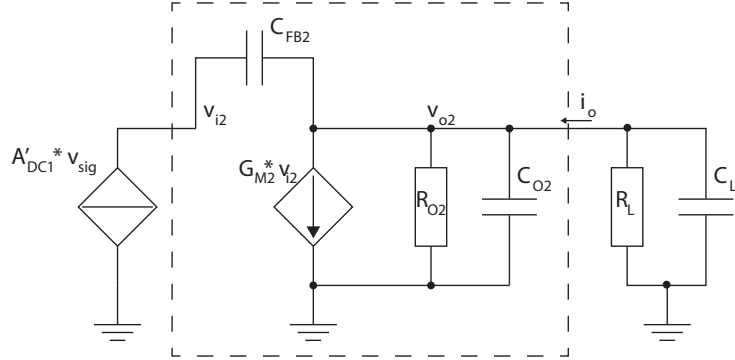


Figure 9.10: Cascode gain stage frequency model assuming dominant pole at output.

get maximal DC gain when R_L matches $R_O \approx g_m r_{ds}^2$, e.g. if R_L is a cascode current source. In that case the input resistance of the CG stage is of the order of $R_{I2} \approx r_{ds}$. Thus the cut off at the input node appears at the same frequency as for a CS stage alone with a single bias transistor as load. So the cascode gain stage gives us its significantly increased gain without loss of BW. On the other hand if we use but a single transistor as load for the cascode gain stage, we have seen earlier that we do 'only' get approximately the gain of a single CS stage again. However, now the BW at the input node corresponds to a CS stage with a very small load resistance since the CG's $R_{I2} \approx \frac{2}{g_m}$. Thus the dominant pole time constant appears at about $R_S(C_{GS} + 2C_{GD})$, i.e. the 'Miller term' $(g_m(r_{ds} || R_L) + 1) * C_{GD}$ is reduced to just $2 * C_{GD}$. So the BW is extended by about the gain of the equivalent single CS stage.

In case b) illustrated in figure 9.10 we are unfortunately not so well off: choosing a big $R_L \approx g_m r_{ds}^2$ will still increase the DC gain A'_{DC} but at the trade off of a proportional decrease in BW. On the other hand, choosing a small $R_L \approx r_{ds}$ will only increase the BW back to the approximate level of a CS stage, as well as the DC gain.

Chapter 10

Current Mirrors

A current mirror (see symbol in figure 10.1) can be seen as a current amplifier with intrinsic current gain A_{I0} , i.e. a circuit receiving an input current I_I and producing an output current I_O that is an amplified copy of that current. The ideal current amplifier has zero input resistance (ideal current sink) and infinite output resistance (ideal current source). It is a three terminal device where both input and output current are conveyed to a third reference node.

In the not so ideal case, one would include an input resistance R_I and an output resistance R_O . Considering a Norton equivalent signal source with an ideal current source i_{sig} and an output resistance R_S in parallel and a load resistance R_L to Gnd at the output one gets:

$$\begin{aligned} i_i &= i_{sig} \frac{R_S}{R_I + R_S} \\ i_o &= A_{I0} i_i \frac{R_O}{R_O + R_L} \\ A_I &:= \frac{i_o}{i_{sig}} = A_{I0} \frac{R_O}{R_O + R_L} \frac{R_S}{R_I + R_S} \end{aligned} \quad (10.1)$$

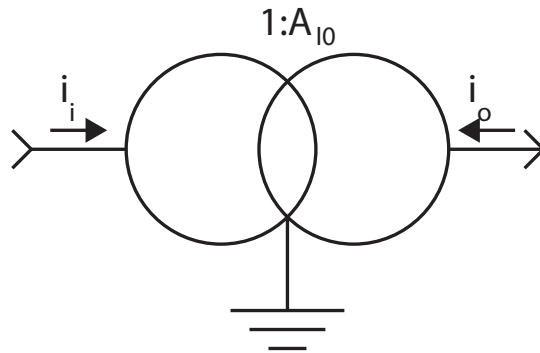


Figure 10.1: Current mirror symbol

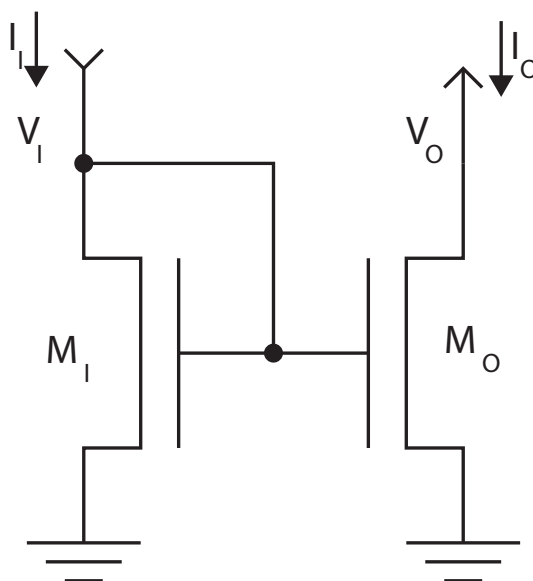


Figure 10.2: A current mirror's simplest two transistor CMOS implementation

10.1 Simple CMOS implementation

A simple implementation with FETs is depicted in figure 10.2. It is close to an ideal current mirror as long as both transistors are in saturation (!). The input transistor M_I is simply a **diode connected FET**. Its gate voltage $V_{GI} = V_I$ (which is also its drain voltage $V_{DI} = V_I$) will converge so that it conveys I_I as its drain current and is shared with the gate of the output transistor M_O . The input transistor will be in saturation as long as the required V_{GSI} (assuming saturation) is in fact bigger than V_{sat} . This is universally the case for all but extremely small I_I : only if $I_I = I_{DI}$ would require a $V_I = V_{GS} < 4U_T$ using its saturation approximation (2.19), only then would there be no consistent solution in the saturation region of operation when given I_I . But again: this only occurs for extremely and usually impractically small input currents. Thus, we do make the assumption for the following musings that:

$$V_I > V_{sat} \quad (10.2)$$

If also M_O is in saturation ($V_O > V_{sat}$), the output current I_O will be very similar to the input current multiplied by the relative width to length ratios:

$$K := \frac{W_O L_I}{L_O W_I} \quad (10.3)$$

So in the large signal world we get

$$I_O \approx K I_I \quad (10.4)$$

10.1.1 Large signal output current offset

A frequent use for CMOS current mirrors is to statically create bias currents for analog circuits, i.e. as **bias generator**. In that context the input will be constant, e.g. a main reference current provided from off chip and a current mirror with multiple output branches will provide different bias currents to various circuits on the CMOS chip. So of interest then is mainly how faithfully the output current reflects the input current with the intended proportionality constant K set by the ratio of the two transistors respective width to length ratios ($K = \frac{W_O L_I}{L_O W_I}$):

$$A_{I0} \approx K := \frac{\frac{W_O}{L_O}}{\frac{W_I}{L_I}} = \frac{W_O L_I}{L_O W_I} \quad (10.5)$$

And consequently one gets:

$$I_O \approx K I_I \quad (10.6)$$

However, dependent on the load one connects to the output, another asymmetry (beyond the intentional different $\frac{W}{L}$) can be the drain voltages at the input- and output transistors. Since the large signal models contain the multiplicative term $(1 + V_{DS}\lambda)$ and assuming both transistors have the same λ (i.e. same length L) we get an **offset error** $\epsilon = I_O - K I_I$ from (10.6):

$$\begin{aligned} K I_I &= K I'_D (1 + V_{DI}\lambda) \\ I_O &= K I'_D (1 + V_{DO}\lambda) \\ I_O - K I_I &= \epsilon = K I'_D (\Delta V_D \lambda) = K \frac{\Delta V_D}{r_{dsI}} = \frac{\Delta V_D}{r_{dsO}} \end{aligned} \quad (10.7)$$

Where $\Delta V_D = V_{DO} - V_{DI}$ and I'_D is the saturation current in M_I without the influence of channel length modulation (i.e. for $\lambda = 0$). You may notice that we use the small signal parameter r_{dsI} here in the large signal domain, but that is OK coincidentally here in this specific case as r_{ds} in general is independent of V_D for the entire saturation region, i.e. not just for a specific point of operation.

So not so surprisingly, making the output a better current source (increasing $R_O = r_{dsO}$) reduces the detrimental influence of ΔV_D on the intentional relationship.

Note that in order to *determine* V_{DI} and V_{DO} here you'll have to use the large signal models appropriate for the situation. That will usually be either the weak or strong inversion models in saturation. Thus, the deduction will be more complicated at the input with an exponential or quadratic dependency of the drain current on the gate voltage and simpler on the output with the linear dependency between the drain voltage and current.

10.1.2 Range of operation

Another property to be wary about is the limited *range* of voltages $V_{DO} > V_{sat}$ at the output transistor M_O drain for which it remains in saturation! In this simple implementation this is a single V_{sat} that has to be appropriately derived from the biasing current I_{DO} according to (2.14). So for example trying to drive

a large load at the output may cause the output voltage to end up smaller than V_{sat} and this implementation of a current mirror will fail to operate properly.

$$V_{DO} \stackrel{!}{>} V_{sat} \quad (10.8)$$

10.1.3 Small signal LF/DC gain

So once we have assured that the simple current mirror's transistors are in saturation and have gauged the offset error, we might not just use the circuit as a constant bias current source but in a larger system as a current amplifier with dynamic input. Thus we are also interested in its small circuit current gain and bandwidth.

We can save ourselves a lot of work by noticing the similarity of the small signal equivalent circuit of the CM in figure 10.3 with that of the general transconductance amplifier in figure 9.4. We can write:

$$A_I = \frac{i_o}{i_{sig}} = \frac{v_o}{v_{sig}} \frac{R_S}{-Z_L} = A_V \frac{R_S}{-Z_L} \quad (10.9)$$

$$= \frac{Z_I}{R_S + Z_I} G_M Z_O \frac{Z_L}{Z_O + Z_L} \frac{R_S}{Z_L} = \frac{Z_I R_S}{R_S + Z_I} G_M \frac{Z_O}{Z_O + Z_L} \quad (10.10)$$

Note that in the general case where $\omega \neq 0$, Z_I and Z_O are complicated frequency dependent expressions that depend on both C_I and C_{FB} , but they simplify to respectively R_I and R_O in the DC case.

Thus, for the DC case, i.e. $\omega = 0$:

$$A_I(0) = \frac{R_S R_I}{R_S + R_I} G_M \frac{R_O}{R_O + R_L} = R_I^* G_M \frac{R_O}{R_O + R_L} \quad (10.11)$$

Now we can insert the appropriate parameters (see table 10.1). Let us first note that the transconductance of the output and input transistors, respectively g_{mO} and g_{mI} are proportional to their respective $\frac{W}{L}$ ratios and to a minor degree also proportional to I_D and thus dependent on V_{DS} , so:

$$g_{mO} \approx K g_{mI} \quad (10.12)$$

and more precisely, the deviation from that intended transconductance is (derived from (10.7)):

$$g_{mO} - K g_{mI} = K I_D' (\Delta V_D \lambda) \quad (10.13)$$

Many publications neglect the influence of ΔV_D as minor, though, thus ignoring 10.13 and treating the equality in 10.12 as precise.

So we have:

$$R_O = r_{dsO} \quad (10.14)$$

$$R_I = r_{dsI} \parallel \frac{1}{g_{mI}} \quad (10.15)$$

Finally, putting everything together:

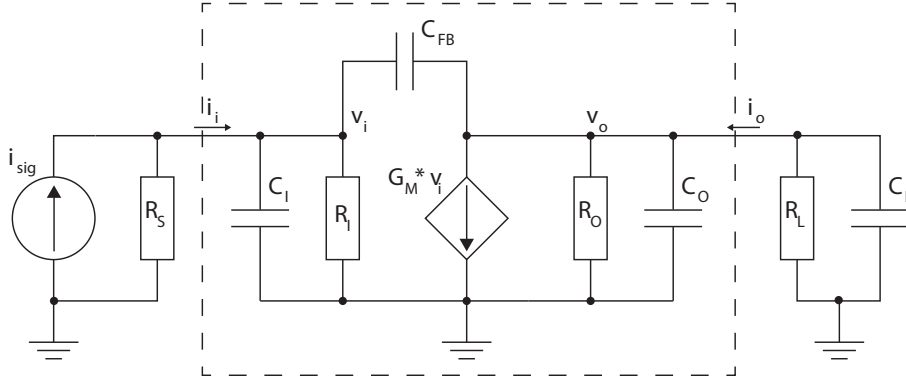


Figure 10.3: A general single stage current amplifier model applicable to the simple CMOS current mirror

$$A_I(0) = (r_{dsI} \parallel \frac{1}{g_{mI}} \parallel R_S) G_M \frac{r_{dsO}}{r_{dsO} + R_L} \approx (r_{dsI} \parallel \frac{1}{g_{mI}} \parallel R_S) K g_{mI} \frac{r_{dsO}}{r_{dsO} + R_L} \quad (10.16)$$

or reorganized:

$$A_I(0) = \frac{i_o}{i_{sig}} = K \underbrace{\frac{R_S}{(r_{dsI} \parallel \frac{1}{g_{mI}}) + R_S}}_{\text{due to } R_S \neq \infty} \underbrace{\frac{g_{mI} r_{dsI}}{1 + g_{mI} r_{dsI}}}_{\text{due to } G_M \neq \frac{1}{R_I}} \underbrace{\frac{r_{dsO}}{r_{dsO} + R_L}}_{\text{due to } R_O \neq \infty} \quad (10.17)$$

So note that the three terms to the right are all close to being 1 and consequently the total current gain is close to being simply

$$A_I(0) \approx K \quad (10.18)$$

and let's just postulate that the desired gain is in fact K , so we treat those deviations as undesirable. So ideally we want $R_S \gg (r_{dsI} \parallel \frac{1}{g_{mI}})$, $g_{mI} r_{dsI} \gg 1$ and $r_{dsO} \gg R_L$

10.1.4 Frequency response

Considering an input capacitance $C_I = C_{GSI} + C_{GSO} + C_{DBI}$ and a feedback capacitance between input and output $C_{FB} = C_{GDO}$ and including R_S , R_L , C_L and $C_O = C_{DBO}$ we get the Kirchhoff equations for nodes v_i and v_o , as well as the relationship of i_o and v_o by considering i_o to be the current flowing from the load impedance:

	R_I	G_M	R_O	C_I	C_{FB}	C_O
CS	∞	g_m	r_{ds}	C_{GS}	C_{GD}	C_{DB}
CM	$r_{dsI} \frac{1}{g_{mI}}$	g_{mO}	r_{dsO}	$C_{GSI} + C_{DBI} + C_{GSO}$	C_{GDO}	C_{DBO}

Table 10.1: Equivalent linear amplifier parameters for CS and CM.

$$\begin{aligned}
v_i \left(\frac{1}{R_S} + sC_I + \frac{1}{r_{ds}} + g_{mI} + sC_{FB} \right) &= v_o sC_{FB} + i_{sig} \\
v_o \left(\frac{1}{R_L} + \frac{1}{r_{ds}} + sC_L + sC_O + sC_{FB} \right) + g_{mO} v_i &= v_i sC_{FB} \\
i_o &= -v_o \left(\frac{1}{R_L} + sC_L \right)
\end{aligned} \tag{10.19}$$

The circuit is depicted in figure 10.3. Now, we could solve these equations for $\frac{i_o}{i_{sig}}$... OR ... we could use a shortcut by observing that the whole circuit is very similar to our general single stage transconductance amplifier model in figure 9.4. In fact, the only difference is, that the signal source is now a Norton equivalent instead of a Thévenin equivalent circuit. So we can recycle all the hard work that went into the analysis of that general circuit yet again. We need the relation of the voltage I/O to know the current I/O:

$$i_{sig} = \frac{v_{sig}}{R_S} \tag{10.20}$$

$$i_o = -\frac{v_o}{Z_L} = -v_o \left(\frac{1}{R_L} + sC_L \right) = -\frac{v_o}{R_L} (1 + sR_L C_L) \tag{10.21}$$

So we can write:

$$\frac{i_o}{i_{sig}} = \frac{v_o}{v_{sig}} \frac{R_S}{-Z_L} \tag{10.22}$$

$Z_L = \frac{1+sR_L C_L}{R_L}$ adds another zero z_{o2} to the current mirror's current transfer function.

$$\tau_{z_{o2}} = \frac{1}{\omega_{z_{o2}}} = R_L C_L \tag{10.23}$$

So we now have a transfer function with two poles (from the general transconductance model (9.24) and two zeros (one from the general transconductance model and another from (10.21))

For the appropriate values of the parameters have a look at table 10.1. The parameters for the common source stage are repeated from tables 9.1 and 9.2 as the ones that are most like the current mirror's. The only difference with a qualitative influence on the frequency response is the input resistance R_I . This results in a much diminished DC voltage gain:

$$\begin{aligned}
\frac{v_o(0)}{v_{sig}(0)} &= -G_M R_L^* \frac{R_I}{R_S + R_I} \\
&= -g_{mO} R_L^* \frac{\frac{1}{g_{mI}} || r_{dsI}}{R_S + \left(\frac{1}{g_{mI}} || r_{dsI} \right)} \\
&\approx -K g_{mI} R_L^* \frac{\frac{1}{g_{mI}}}{R_S + \frac{1}{g_{mI}}} \\
&\approx -K \frac{R_L^*}{R_S}
\end{aligned} \tag{10.24}$$

The small R_I also compensates for the Miller effect and thus the pole at the input node is moved to much higher frequencies than it would be for a CS stage. The time constant τ_i at the input node using the simplified approach that assumes that the pole at the *input* node is dominant according to (9.34), yields:

$$\begin{aligned}
\tau_{p_i} &= \frac{1}{\omega_{p_i}} = \left(R_S || \frac{1}{g_{mI}} || r_{dsI} \right) (C_I + C_{FB}(g_{mO} R_L^* + 1)) \\
&\approx \frac{1}{g_{mI}} (C_I + C_{FB}(K g_{mI} R_L^* + 1)) \\
&\approx \frac{C_I}{g_{mI}} + K R_L^* C_{FB} = \frac{C_I}{g_{mI}} + K(r_{dsO} || R_L) C_{FB}
\end{aligned} \tag{10.25}$$

So unlike in a CS stage, the Miller capacitance is not multiplied with R_S but with the usually much smaller $\frac{1}{g_{mI}}$. This makes it much more likely that there is no dominant pole at the input, but that it's more likely found in the dynamics of the output node. However, for later reference (see the current mirror cut off time constant in a current mirror loaded transconductance amplifier in section 11.3.5), if the output is simply grounded in the small signal world (i.e. connected to an ideal voltage source in the large signal world), then there is no dynamics/poles/zeros appearing on the output node voltage, R_L and C_L are zero and the entire transfer function would be simply reduced having just the one single pole at the input $\tau_{p_i} \approx \frac{C_I}{g_{mI}}$.

Using, on the other hand, the simplifying assumption of the dominant pole being at the *output* for derivation of a pole and zero at the output according to (9.27) yields time constants τ_{p_o} and $\tau_{z_{o1}}$:

$$\tau_{p_o} = \frac{1}{\omega_{p_o}} = R_L^* (C_{FB} + C_L^*) = (r_{dsO} || R_L) (C_{FB} + C_L + C_O) \tag{10.26}$$

$$\tau_{z_{o1}} = \frac{1}{\omega_{z_{o1}}} = \frac{C_{FB}}{g_{mO}} \approx \frac{C_{FB}}{K g_{mI}} \tag{10.27}$$

An interesting side note is that the zero z_{o2} from (10.23) and p_o from above's (10.26) will by themselves (if they are the dominant ones among the poles and zeros, this can be visualized on the magnitude Bode plot) reduce the gain by a factor $\frac{\tau_{z_{o2}}}{\tau_{p_o}}$ between frequencies ω_{p_o} and $\omega_{z_{o2}}$.

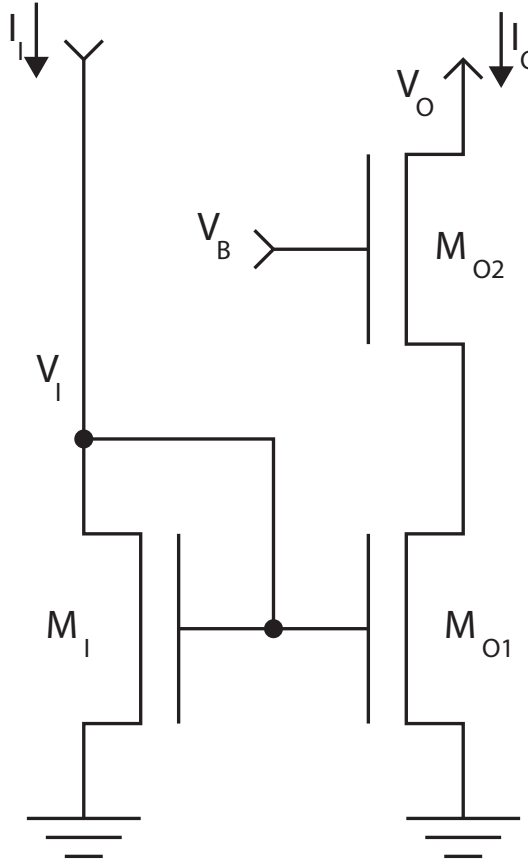


Figure 10.4: Current mirror with cascode output

$$\frac{\tau_{zo2}}{\tau_{po}} = \frac{R_L C_L}{(r_{ds} || R_L)(C_{FB} + C_L + C_O)} = \frac{R_L + r_{ds}}{r_{ds}} \frac{C_L}{(C_{FB} + C_L + C_O)} \quad (10.28)$$

This just marks the transition from the output voltage causing the division of the current between internal and output current being governed by the ratio of internal- and load *resistance* at DC, to being governed entirely by the ratio of internal- and load *capacitances* at very high frequencies.

z_{o1} will usually occur at much higher frequencies (as usually $\frac{1}{g_{mI}} \ll (r_{ds} || R_L)$) and marks a point at which the current will be dominated by the feed forward current through C_{FB} . This will not lead to the current growing to infinity though, since there will be a pole at the input, even though it has been assumed non-dominant here for the simplified deduction of the output node dynamics.

10.2 Cascode output

We noticed the similarity of the simple CMOS implementation to the CS stage. Similarly, if we add a cascode transistor to the output, the resulting circuit is similar to the cascode gain stage with an added small input resistance.

Also using the approximate transconductance (9.17) and output resistance (9.16) of the cascode stage, the only difference to the CS for the DC (!) analysis is the enhanced output resistance. So we get the same result as in (10.16) just substituting the output resistance for the cascode stage (assuming M_{O1} and M_{O2} have the same dimensions):

$$A_I(0) \approx (r_{dsI} || \frac{1}{g_{mI}} || R_S) K g_{mI} \frac{g_{mO} * r_{dsO}^2}{g_{mO} * r_{dsO}^2 + R_L} \quad (10.29)$$

So enhancing the output resistance substantially (compare to (10.16)), makes the current mirror output much less dependent on the load resistance R_L .

However, there is a drawback as well: the point of operation output current I_O may suffer a bigger offset from the input biasing current I_I . Simply put, this is caused by the output branch now being less similar to the input branch. Mostly there can be a big difference between the drain voltages of M_I and M_{O1} now. This depends much on the choice of the bias voltage V_B on the gate of M_{O2} that will draw its source after the gate (in the manner of a SF circuit), and that source is at the same time the drain of M_{O1} .

Another drawback is the reduced output voltage range required to keep all transistors in saturation. For the simple two transistor CMOS implementation it was required that $V_O > V_{sat}$. Here we need:

$$\begin{aligned} V_{DO2} &\stackrel{!}{>} V_B - V_{GS} + V_{sat} \\ V_B &\stackrel{!}{>} V_{GS} + V_{sat} \end{aligned} \quad (10.30)$$

So when the minimum value for V_B is chosen the output voltage needs to be:

$$V_{DO2} \stackrel{!}{>} 2V_{sat} \quad (10.31)$$

10.3 Cascode at output and input

Adding a copy of the same cascode transistor (figure 10.5) to the input branch may appear like a good idea in order to increase the symmetry and to reduce the danger of an offset of the DC current. However, that does not really work out as V_{DI1} needs to be V_{GS} if M_{I1} is to be in saturation. Thus, if $V_B > V_{GS} + V_{sat}$, then the *actual* V_{GS12} is bigger than our 'saturation V_{GS} ' and thus the cascode transistor M_{I2} cannot be in saturation. Consequently that transistor at the input branch does not really help.

However, this initial idea can be fixed by actually connecting the gate of M_{I1} to the input node instead of its own drain as shown in figure 10.6. This circuit helps a lot to improve the large signal offset. If you consider the cascode transistors to act like SFs and thus to force the drains of M_{I1} and M_{O1} to be

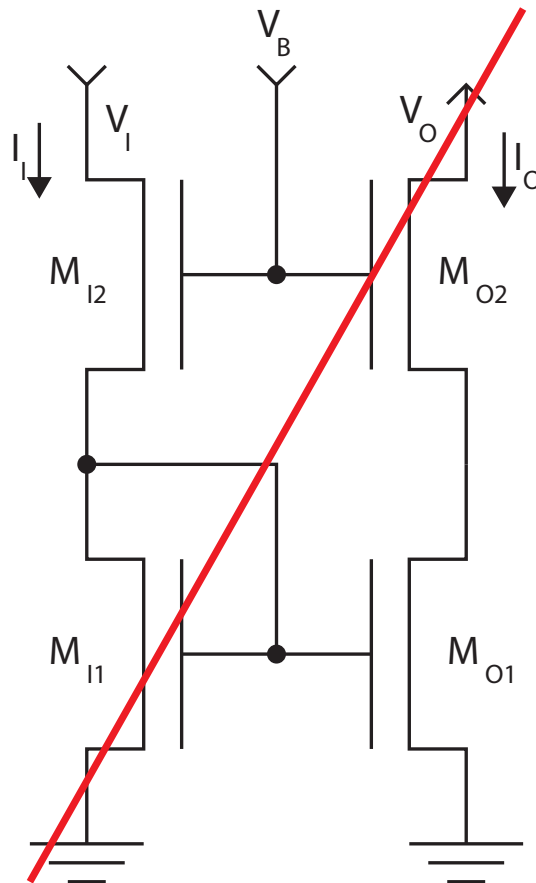


Figure 10.5: An initial and maybe obvious idea for a Cascode current mirror with a cascode at the input as well as the output and manual cascode bias, that unfortunately DOES NOT WORK.

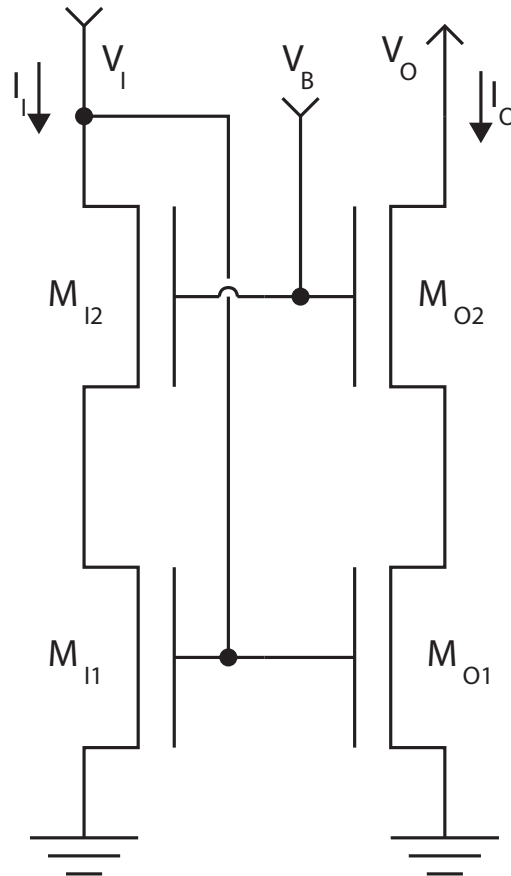


Figure 10.6: An improved Cascode current mirror with a cascode at the input as well as the output and manual cascode bias. It is functional, but has restrictions on both the cascode bias and the maximal input current that one need to be aware of at design (!) time.

the same, they do now have the same voltages on all terminals and will thus conduct the same current if they have the same dimensions, or currents entirely proportional to K if they only have different widths W but the same length L .

In the following we will assume that all transistors have the same dimensions W and L and that thus $K = 1$ to keep the discussion simple.

Note that the range of input currents has now a further limitation: assuming all transistors ARE in saturation V_{GS} for M_{I1} and M_{O1} is determined by the current. In order for the assumption that M_{I1} and M_{O1} are in saturation to hold, we must choose:

$$V_B \stackrel{!}{>} V_{sat} + V_{GS} \quad (10.32)$$

as before. And as before the drain of M_{O2} , but now also the drain of M_{I2} , needs to be

$$\begin{aligned} V_{DO2} &\stackrel{!}{>} V_B - V_{GS} + V_{sat} \\ V_{DI2} = V_{GS} &\stackrel{!}{>} V_B - V_{GS} + V_{sat} \end{aligned} \quad (10.33)$$

Consequently V_B must be chosen carefully within a given range, and can neither be too small nor too big for a given bias current:

$$2V_{GS} - V_{sat} \stackrel{!}{>} V_B \stackrel{!}{>} V_{sat} + V_{GS} \quad (10.34)$$

Again, as before when one is able to tune V_B to be close to its minimum allowed value one can achieve the maximum range of output voltages:

$$V_{DO2} \stackrel{!}{>} 2V_{sat} \quad (10.35)$$

Furthermore, besides the limitations for V_B there is also a limitation for V_{GS} and thus the corresponding I_I . Since $V_{GS} = V_{DI2}$ we get:

$$V_{GS} \stackrel{!}{>} 2V_{sat} \quad (10.36)$$

And this may not be possible if both V_{sat} and V_{GS} are determined by I_I , as for example in strong inversion we have

$$\begin{aligned} V_{GS} = V_t + V_{OV} &\stackrel{!}{>} 2V_{sat} = 2V_{OV} \\ \Rightarrow V_t &\stackrel{!}{>} V_{OV} \end{aligned} \quad (10.37)$$

Thus setting an **upper limit for the current** that this circuit works for. That is, for an optimal V_B : the current is even more limited when choosing a bigger than minimum V_B . Thus one needs to design this current mirror carefully and choose appropriate W and L at *design time* (!!!) to target a specific limited current range. So the circuit is not to be used in applications, where the current range is unknown beforehand. Typical use cases are in cascode amplifiers where the bias current is chosen at design time.

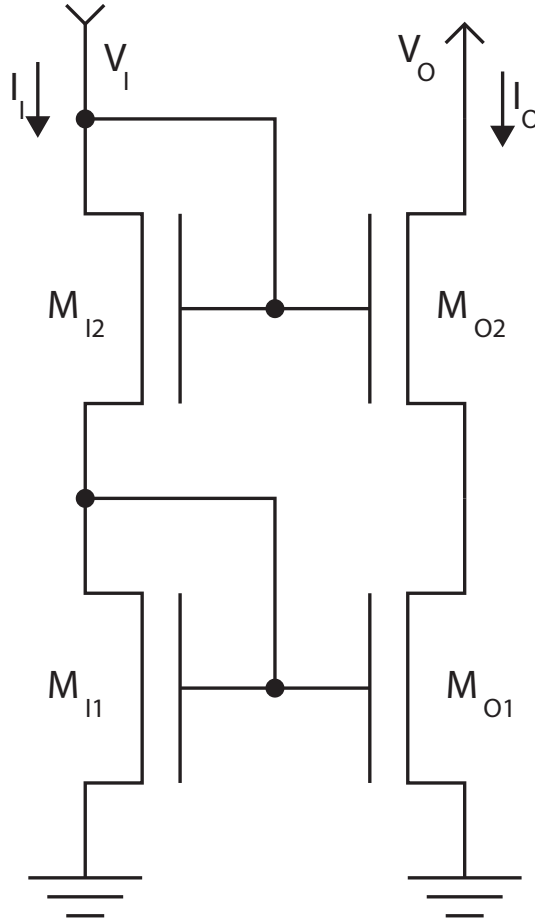


Figure 10.7

10.4 Automatic biasing of output cascode

Both above variants with cascode outputs have the distinct disadvantage of requiring an external bias voltage for the cascode transistor that needs scaling to the actual current point of operation. It would be more convenient if said bias voltage could actually be generated automatically and there are indeed current mirror variants that do that.

10.4.1 Double diode input, cascode output

A simple idea is the double diode input (see figure 10.7). The second diode connected transistor M_{I2} generates an appropriate gate voltage for itself and the output cascode transistor M_{O2} . The output works just as for the cascode output CM with an external V_B at the gate of M_{O2} . A potential drawback is that while an allowed value for V_B is chosen, it is not the minimal allowed V_B and thus the output voltage range is not maximized. The gate voltage at M_{I2}

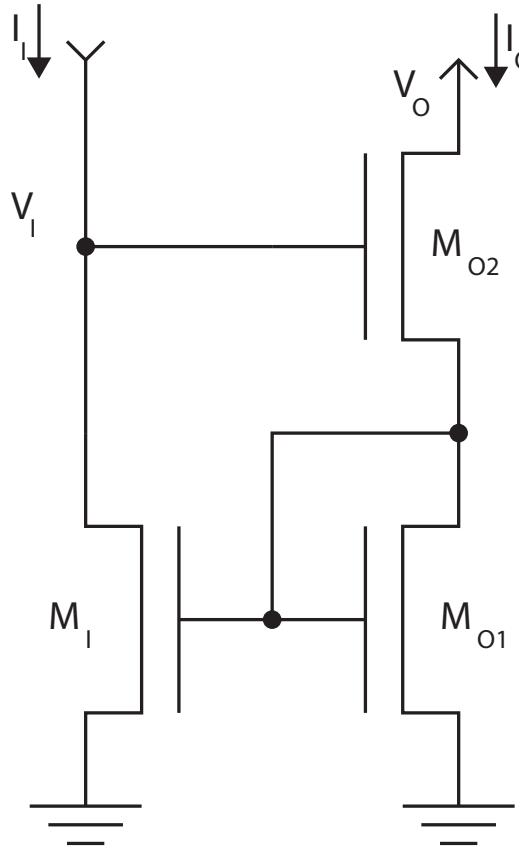


Figure 10.8: Wilson Current mirror

will settle at:

$$V_{DI2} = V_{GI2} = V_{GO2} = 2V_{GS} \quad (10.38)$$

and thus the output voltage needs to be bigger than:

$$V_{DO2} \stackrel{!}{>} V_{GS} + V_{sat} \quad (10.39)$$

and since $V_{GS} > V_{sat}$, the minimum output voltage is bigger than it could be for the versions with the 'manually' tuned V_B . This is unfortunately the same trade off for all three CMs with automatic output cascode tuning discussed here. They will settle on the same gate voltage for the output cascode transistor, or more generally on the same voltages on all nodes at the output branch.

10.4.2 The Wilson CM

This variant will settle on the same voltages on the output branch by a somewhat more convoluted route with significant feedback between the input and output branch. Just assuming that there is a fixed point on which all voltages in the

circuit will settle so that all transistors are in saturation and conduct the same current I_I (ignoring channel length modulation and back gate effects, as we have done throughout for the current mirror analyses), yields that the gates of M_{I1} and M_{O1} must be at V_{GS} and consequently the gate of M_{O2} which is shorted to the drain of M_{I1} must be at $2V_{GS}$ and thus its drain must be above $V_{GS} + V_{sat}$. This value also assures that the drain voltage of M_{I1} puts it into its saturation region, and thus this CM settles on the same input voltage as the double diode input variant.

One is saving one transistor in this design, but there is substantial feedback going on between the output and input branch before the circuit settles and stability analysis is more challenging than for the double diode input cascode CM. Furthermore, the drain voltages on M_{I1} and M_{O2} are respectively at $2V_{GS}$ and V_{GS} , so quite different and will cause some large signal offset error. On the other hand, while they are different, their ratio is well defined and is largely independent from the output voltage, so said offset error can be predicted well and is largely independent of the output load.

10.4.3 The modified Wilson CM

This variant depicted in figure 10.9 can be seen as an extension of the Wilson CM or a changed double diode input variant. In fact, all transistor terminal voltages will settle on the same voltages as the double diode input variant, but once again with a somewhat stronger feedback component, so again the stability analysis is more involved, and it might not be the preferred choice when used with a dynamically changing input current. For constant input currents (i.e for biasing circuits) though, it is an alternative choice to the double diode input.

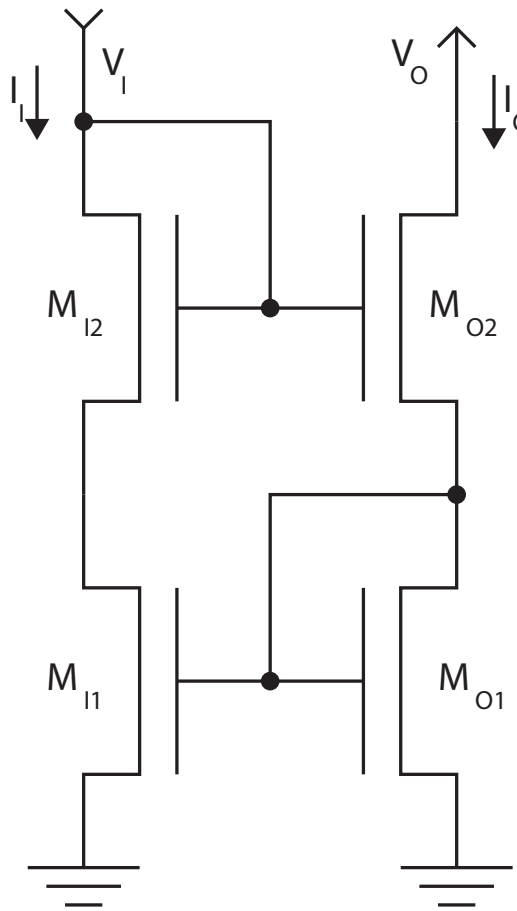


Figure 10.9: The modified Wilson CM

Chapter 11

Differential Amplifiers

Differential amplifiers (i.e. with differential input, while 'fully differential' amplifiers also have differential output, see Fig. 11.1) refer their input not to global $-V_{ss}$ or Gnd but to an explicit reference input V_- . This makes the differential amplifier more flexible than the single ended variants we have looked at so far. Advantages are immunity to correlated noise at the two inputs and the possibility to use them in feedback circuits, as well as rendering the concept of negative inputs simple.

11.1 Differential pair

FET differential amplifiers are usually based on the differential pair. Its core are two transistors in parallel ideally connected to a current source at their sources, but realistically this is usually a third transistor with a fixed bias voltage at its gate and thus a good approximation of a good current sources with high output resistance. See figure 11.2.

The intuitive explanation of its function is that the total current is limited and divided between the two parallel transistors. That balance tips rapidly in favour of the transistor with the higher gate voltage input and is largely only dependent on the *difference* of the input voltages and only to a minimal degree by the absolute input values. So in the small region where the inputs are close and both branches receive part of the bias current the circuit acts like a **differential transconductance amplifier** that linearly splits the total current proportionally to the difference of the input voltages. And if the two inputs are different enough that the entire bias current just flows through one

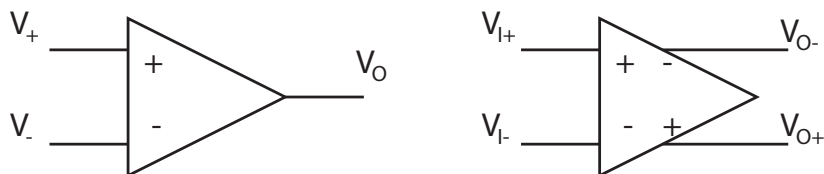


Figure 11.1: Differential amplifier symbols

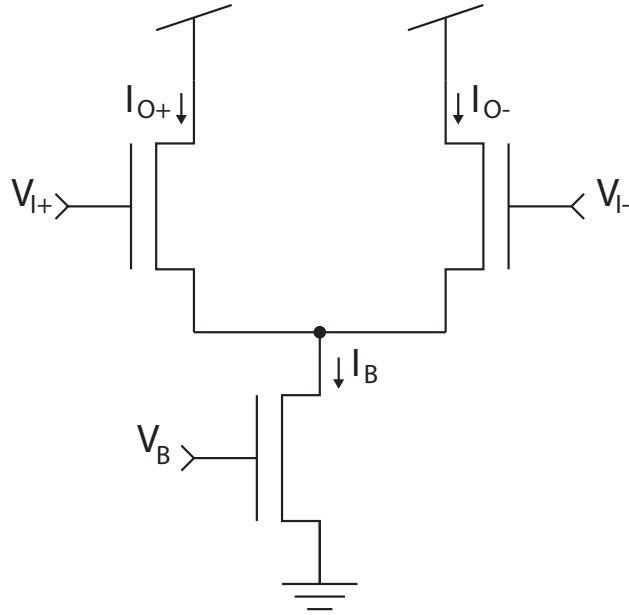
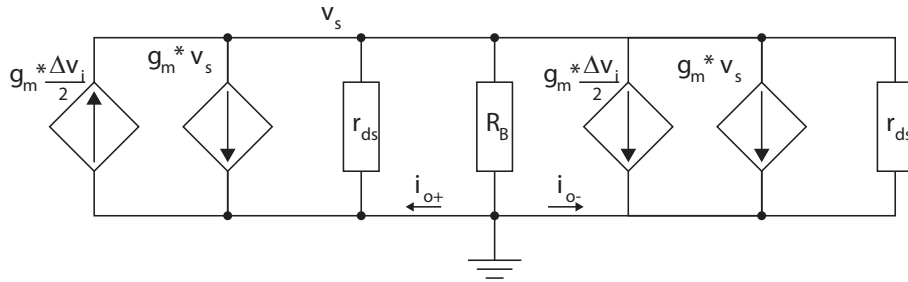
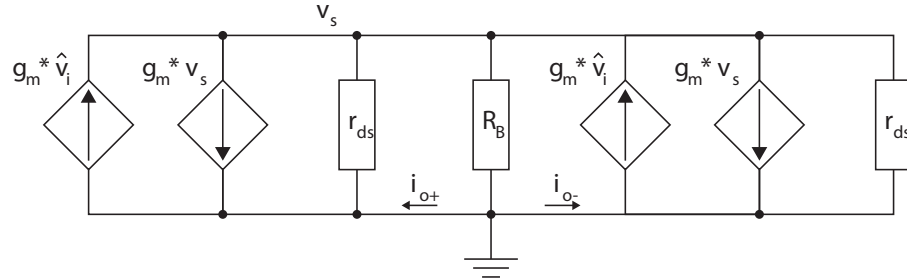


Figure 11.2: Differential pair basics with differential current output

Figure 11.3: Small signal differential pair for balanced differential input Δv_i with differential current output.Figure 11.4: Small signal differential pair for common mode input Δv_i with differential current output.

branch, the circuit acts as a **comparator** that compares two analog inputs and turns them into a digital output. Comparators are for instance essential components of analog to digital converters (ADC).

If we consider its operation as a differential transconductance amplifier (making sure we are close to the correct large signal point of operation/biasing point, where each branch conducts exactly half of the bias current!) figure 11.3 shows the small signal equivalent. Note that the voltage controlled current source of the parallel transistors has been separated into two separate sources for the negative and positive contributions of the two control terminals (i.e. gate and source). As we do have two inputs now, let's look at two separate scenarios of correlated changes in these inputs, that we can linearly combine to reflect any change of the two inputs: Let's linearly separate all possible inputs into:

$$\begin{aligned} v_{i+} &= \hat{v}_i + \frac{\Delta v_i}{2} \\ v_{i-} &= \hat{v}_i - \frac{\Delta v_i}{2} \end{aligned} \quad (11.1)$$

where we define the average of the input voltages as the **common mode input**

$$\hat{v}_i := \frac{v_{i+} + v_{i-}}{2} \quad (11.2)$$

and the difference of the input voltages as **differential input**

$$\Delta v_i := v_{i+} - v_{i-} \quad (11.3)$$

So in the manner of linear systems, we can investigate the effects of changes of Δv_i and $\hat{v}_i$ separately and treat arbitrary changes as a linear combination of the two separate effects.

Let's first consider what happens if only the difference of the inputs Δv_i changes but their average remains constant, i.e. the plus terminal receives plus half the difference and the negative terminal minus half the difference. Note in figure 11.3, instead of writing a minus sign at the negative terminal input the direction of the current has been inverted. And finally the resistor R_B is the output resistance of the bias current source, so in the case of a FET it will be that transistor's r_{ds} .

If we write down the Kirchhoff current equations for node v_s and for the two output currents i_{o+} and i_{o-} we get:

$$\begin{aligned} i_{o+} + v_s \left(g_m + \frac{1}{r_{ds}} \right) &= g_m \frac{\Delta v_i}{2} \\ i_{o-} + v_s \left(g_m + \frac{1}{r_{ds}} \right) &= -g_m \frac{\Delta v_i}{2} \\ v_s \left(2g_m + 2\frac{1}{r_{ds}} + \frac{1}{R_B} \right) &= \frac{\Delta v_i}{2} (g_m - g_m) = 0 \end{aligned} \quad (11.4)$$

You notice that the current being balanced that way means that the only small signal currents depending on the input Δv_i flowing into node v_s are from

the two differential current sources that cancel each other out. Consequently, v_s will be constant in this scenario, i.e. $v_s = 0$ (compare the last line of (11.4)). We get the result for the differential output current by subtracting the first two lines of (11.4).

$$\left(i_{o+} + v_s \left(g_m + \frac{1}{r_{ds}}\right)\right) - \left(i_{o-} + v_s \left(g_m + \frac{1}{r_{ds}}\right)\right) = 2g_m \frac{\Delta v_i}{2} \quad (11.5)$$

$$\Delta i_o = g_m \Delta v_i$$

Let's now have a look at what happens if one changes only the **common mode input voltage**, which we defined as the average of the two inputs $\hat{v}_i := \frac{v_{i+} + v_{i-}}{2}$. See figure 11.4. The Kirchhoff current equations in this case are:

$$i_{o+} = i_{o-} = \hat{v}_i g_m - v_s \left(g_m + \frac{1}{r_{ds}}\right) \quad (11.6)$$

$$v_s \left(2g_m + 2\frac{1}{r_{ds}} + \frac{1}{R_B}\right) = 2\hat{v}_i g_m$$

Resulting in

$$i_{o+} = i_{o-} = \hat{v}_i \frac{\frac{g_m}{R_B}}{2g_m + 2\frac{1}{r_{ds}} + \frac{1}{R_B}} \quad (11.7)$$

So the output currents are changed together in the same direction as $\hat{v}_i$. In the small signal world, this does not affect the differential output current at all because of the superposition principle of linear systems: one can change $\hat{v}_i$ and Δv_i separately or together and the result at the output is just a superposition of the individual effects. So the differential output result will still not change even if $\hat{v}_i$ is different.

However, looking at the situation with the large signal spectacles, changing the output currents together in the same direction means changing the voltage V_S and thus the bias current I_B . Remember that bias parameters are those that set the point of operation aka biasing point. The effect here will be that the small signal parameters have to be re-evaluated: in fact g_m depends on the bias current and will change! So the result of a change of $\hat{V}_I$ (note the use of the large signal parameter here!) will lead to a change in transconductance, i.e. will CHANGE THE GAIN of the differential pair. These common mode gain errors are NOT COVERED by the usual derivation of the widely used **common mode rejection ratio (CMRR)** which we will discuss later.

11.2 R-load gain

A first extension that allows to read out a differential voltage from the differential output currents of the differential pair can be achieved by adding resistive loads to both branches (figure 11.5).

Deriving the Kirchhoff equations yields:

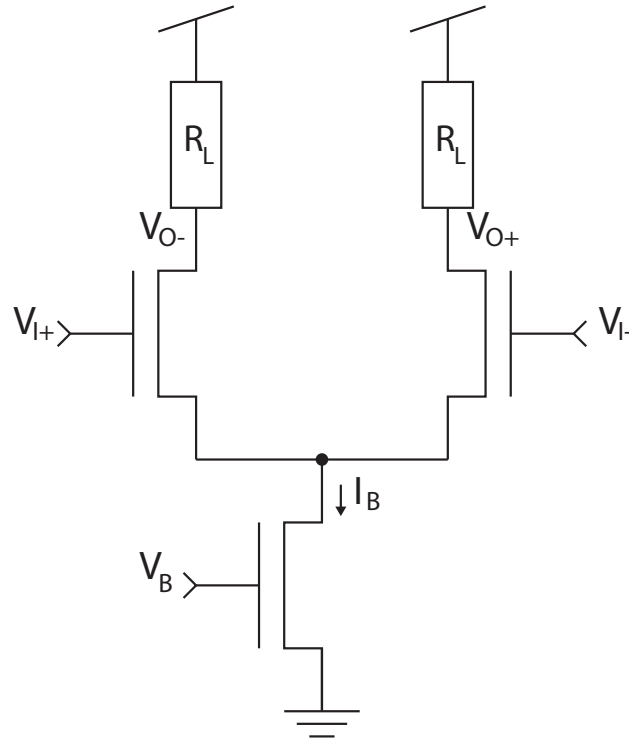


Figure 11.5: Differential pair with resistors as load on each branch and thus differential voltage output.

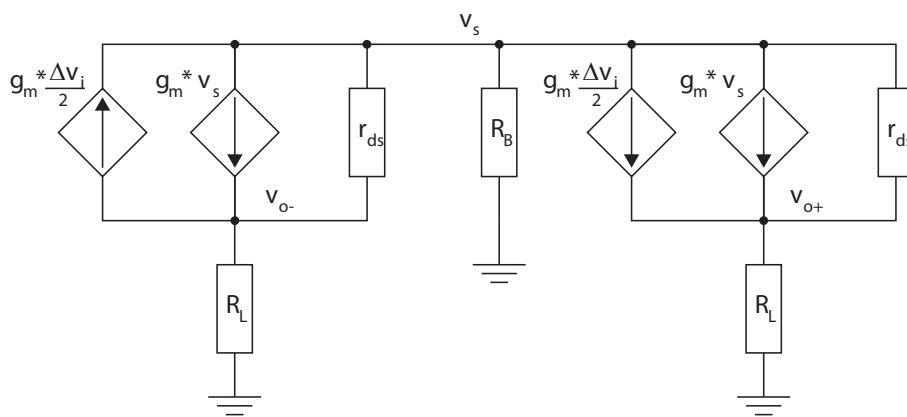


Figure 11.6: Small signal differential pair for balanced input Δv_i with load resistors and differential voltage output.

$$\begin{aligned}
v_{o+} \left(\frac{1}{R_L} + \frac{1}{r_{ds}} \right) - g_m * \frac{\Delta v_i}{2} - g_m v_s &= v_s \frac{1}{r_{ds}} \\
v_{o-} \left(\frac{1}{R_L} + \frac{1}{r_{ds}} \right) + g_m * \frac{\Delta v_i}{2} - g_m v_s &= v_s \frac{1}{r_{ds}} \\
v_s \left(\frac{1}{R_B} + \frac{2}{r_{ds}} + 2g_m \right) &= v_{o+} \frac{1}{r_{ds}} + v_{o-} \frac{1}{r_{ds}}
\end{aligned} \tag{11.8}$$

And solving them (one can for example again use the short cut that $v_s = 0$ for balanced differential input and subtract the first two lines):

$$\begin{aligned}
(v_{o+} - v_{o-}) \left(\frac{1}{R_L} + \frac{1}{r_{ds}} \right) - 2g_m * \frac{\Delta v_i}{2} &= 0 \\
\frac{\Delta v_o}{\Delta v_i} &= g_m (R_L || r_{ds})
\end{aligned} \tag{11.9}$$

We will spare ourselves the detailed analysis of the effects of the common mode input $\hat{v}_i$: by the sheer symmetry of the schematics it is kind of trivial that when changing both inputs, one will change the output voltages but crucially: *not their difference*. So like before one need not worry about effects of the common mode input in the *small signal world*, but again when looking at large changes of the average input voltages, one needs to consider a different point of operation, and ergo a different bias current and g_m , and consequently the *gain* will change in dependence of the large signal average input voltage level!

11.3 Differential transconductance amplifier

Maybe the most common basic way of implementing a differential amplifier with a single ended output uses the differential pair with a current mirror also known as **transconductance amplifier** (Fig. 11.7)¹. The current mirror mirrors the output current from the left branch of the differential pair into the right branch from the top. So the right branch drain voltage is pulled high, when the left branch wins and gets virtually all the bias current. Vice versa if the left branch gets all the current the current mirror injects no current at all and the right hand side drain voltage is pulled low. And in the interesting case where the input voltages are close enough to each other and *both* branches get a share in the bias current, the circuit behaves like an amplifier, i.e. the difference of their shares is conveyed to the output. Then, with the circuit's output resistance and possibly a load resistance in parallel, that differential current is turned into a proportional output voltage.

11.3.1 Differential gain

The small signal equivalent circuit is depicted in figure 11.8 which gives rise to the Kirchhoff current equations in (11.10)

¹The name 'transconductance' amplifier stems from the fact that the circuit (just like the single ended CS gain stage) has a high output resistance and is thus a good current source, providing a current output that is only mildly affected by a load resistance. When the output is left open circuit or connected to a very large load resistance this results in a high voltage gain, i.e. the current output multiplied with the high output resistance.

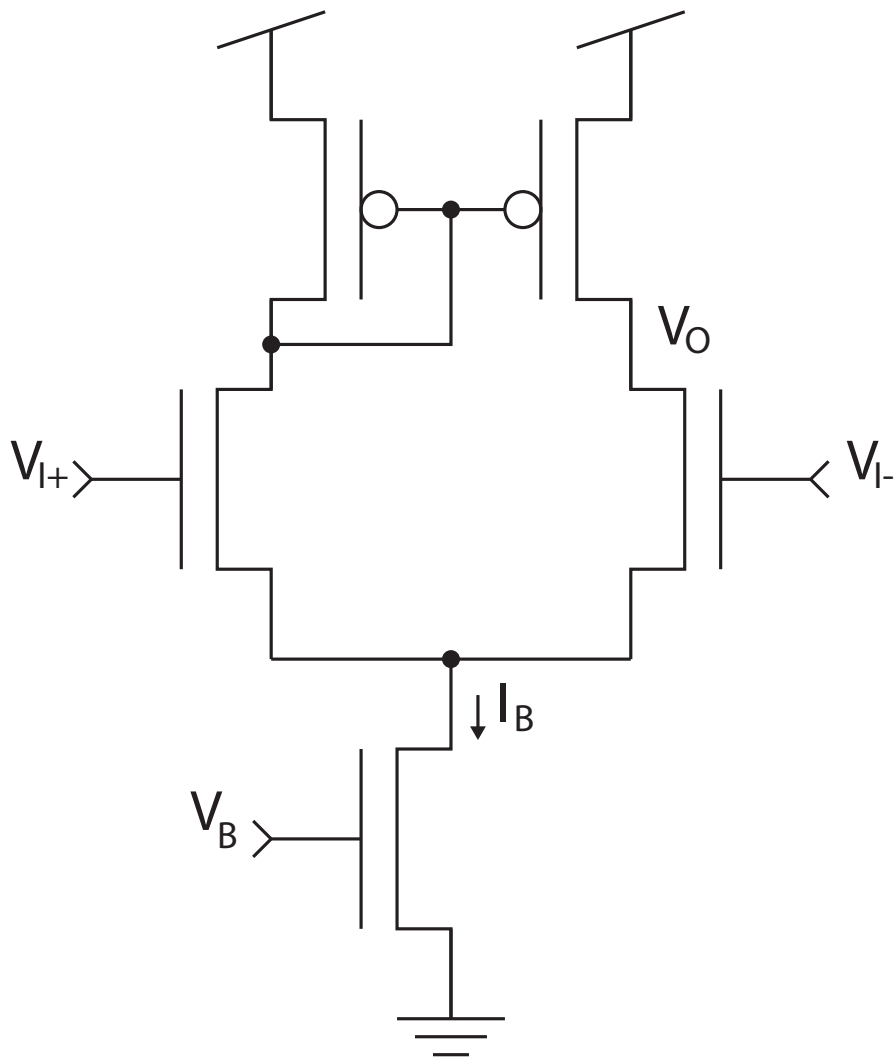


Figure 11.7: The differential pair with a current mirror load, also known as basic differential transconductance amplifier. It mirrors the left hand side current to the right hand side branch. Thus the current in the branches will compete against each other in the output node, either pulling the output high or low.

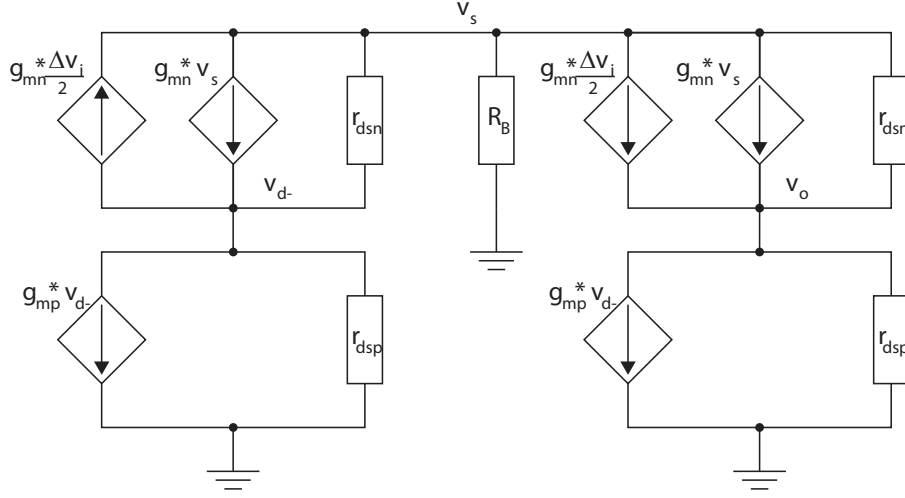


Figure 11.8: The small signal equivalent model for a differential pair with a current mirror load that mirrors the left hand side current to the right hand side branch (Fig. 11.7). Note that the branches are 'folded down' where the lower Gnd terminals correspond to the Vdd terminals in figure 11.7.

$$\begin{aligned}
 v_d \left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} + g_{mp} \right) + \frac{\Delta v_i}{2} g_{mn} &= v_s \left(g_{mn} + \frac{1}{r_{dsn}} \right) \\
 v_s \left(2 \frac{1}{r_{dsn}} + 2g_{mn} + \frac{1}{R_B} \right) &= v_d \frac{1}{r_{dsn}} + v_o \frac{1}{r_{dsp}} \quad (11.10) \\
 v_o \left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} \right) + v_d g_{mp} &= v_s \left(g_{mn} + \frac{1}{r_{dsn}} \right) + \frac{\Delta v_i}{2} g_{mn}
 \end{aligned}$$

Solving this completely² results in equation 11.11:

$$\frac{v_o}{\Delta v_i} = g_{mn} (r_{dsn} || r_{dsp}) \frac{2g_{mp}g_{mn} + 2\frac{g_{mp}}{r_{dsn}} + \frac{g_{mp}}{R_B} + \left(\frac{g_{mn}}{r_{dsp}} + \frac{1}{r_{dsn}r_{dsp}} + \frac{1}{2r_{dsn}R_B} + \frac{1}{2r_{dsp}R_B} \right)}{2g_{mn}g_{mp} + 2\frac{g_{mp}}{r_{dsn}} + \frac{g_{mp}}{R_B} + 2 \left(\frac{g_{mn}}{r_{dsp}} + \frac{1}{r_{dsp}r_{dsn}} + \frac{1}{2r_{dsn}R_B} + \frac{1}{2r_{dsp}R_B} \right)} \quad (11.11)$$

The large division term to the right is close to 1 but slightly smaller, if we assume that $\frac{1}{R_S} \approx \frac{1}{r_{dsn}} \approx \frac{1}{r_{dsp}} \ll g_{mn} \approx g_{mp}$. That's a reasonable assumption if we start from assuming that a basic CS gain stage biased the same as the transistors in the differential transconductance amplifier gets us a gain of, for example, at least 10, then $g_m r_{ds} > 10 \Rightarrow \frac{g_m}{10} > \frac{1}{r_{ds}}$. So if we assume that all resistors are the same and that g_m is at least 10 times larger than $\frac{1}{r_{ds}}$ then (11.11) gives us:

²Not something I recommend repeating ;-). I took me some days to do and recheck several times. I should definitely have gotten some symbolic math software, like Maple. Let me know if you find an error!

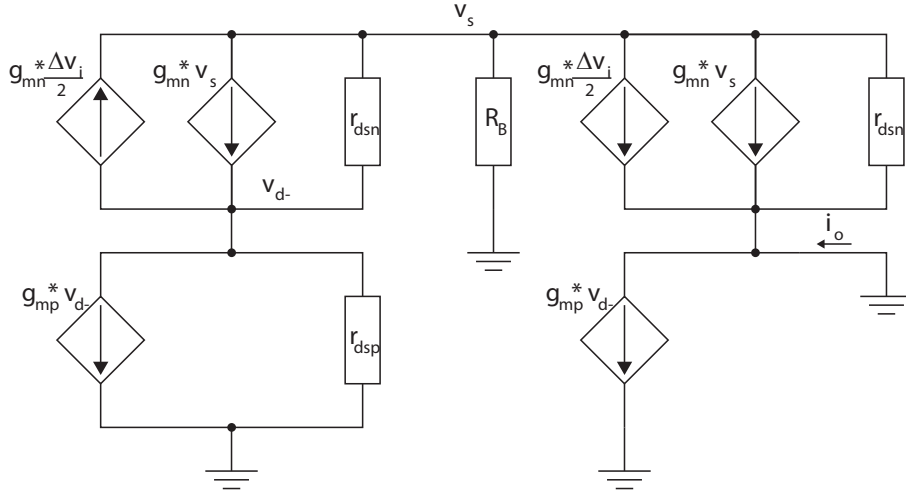


Figure 11.9: The small signal equivalent model for a differential pair with a current mirror load (Differential Transconductance Amplifier), with balanced differential input and the output at a fixed voltage to deduce the total transconductance G_M . Giving rise to (11.13).

$$\begin{aligned}
 g_{mn}(r_{dsn} || r_{dsp}) &> \frac{v_o}{\Delta v_i} > g_{mn}(r_{dsn} || r_{dsp}) \frac{200 + 40 + 2}{200 + 50 + 4} = g_{mn}(r_{dsn} || r_{dsp}) * 0.953 \\
 &\Rightarrow \\
 \frac{v_o}{\Delta v_i} &\approx g_{mn}(r_{dsn} || r_{dsp})
 \end{aligned} \tag{11.12}$$

So for all practical purposes, we shall assume that the gain is exactly the same as for a CS gain stage: $\frac{v_o}{\Delta v_i} \approx g_{mn}(r_{dsn} || r_{dsp})$. Note that any external load R_L would need to be added in parallel with $(r_{dsn} || r_{dsp})$!

11.3.2 G_M and R_O deduced separately

An alternative deduction of the same result for the gain is to deduce expressions for G_M and R_O separately first and then to multiply them to get $A = G_M R_O$. This approach is also documented here, since it might come in handy to also know the parameters G_M and R_O in other contexts.

Let's first deduce G_M . From figure 11.9:

$$\begin{aligned}
 v_d \left(\frac{1}{r_{dsp}} + \frac{1}{r_{dsn}} + g_{mp} \right) + \frac{\Delta v_i}{2} g_{mn} &= v_s \left(\frac{1}{r_{dsn}} + g_{mn} \right) \\
 v_s \left(2g_{mn} + 2\frac{1}{r_{dsn}} + \frac{1}{R_B} \right) &= v_d \frac{1}{r_{dsn}} \\
 v_d g_{mp} &= v_s \left(\frac{1}{r_{dsn}} + g_{mn} \right) + \frac{\Delta v_i}{2} g_{mn} + i_o
 \end{aligned} \tag{11.13}$$

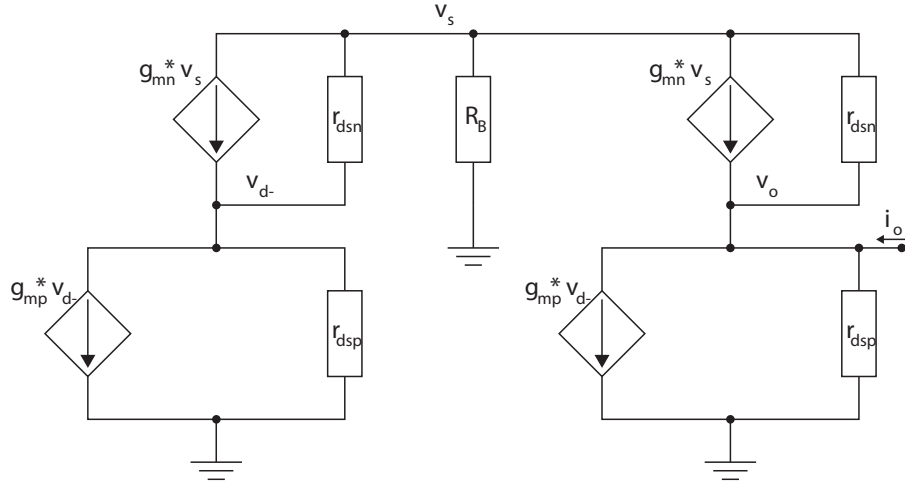


Figure 11.10: The small signal equivalent model for a differential pair with a current mirror load (Differential Transconductance Amplifier), with the input at a fixed voltage to deduce the total output resistance RO . Giving rise to (11.17).

Solving this yields:

$$\begin{aligned}
 G_M &= \frac{i_o}{\Delta v_i} \\
 &= -g_{mn} \frac{2g_{mn}g_{mp} + 2\frac{g_{mp}}{r_{dsn}} + \frac{g_{mp}}{R_B} + \left(\frac{g_{mn}}{r_{dsp}} + \frac{1}{r_{dsn}r_{dsp}} + \frac{1}{r_{dsn}2R_B} + \frac{1}{r_{dsp}2R_B}\right)}{2g_{mn}g_{mp} + 2\frac{g_{mp}}{r_{dsn}} + \frac{g_{mp}}{R_B} + 2\left(\frac{g_{mn}}{r_{dsp}} + \frac{1}{r_{dsn}r_{dsp}} + \frac{1}{r_{dsn}2R_B} + \frac{1}{r_{dsp}2R_B}\right) + \frac{g_{mn}}{r_{dsn}} + \frac{1}{r_{dsn}^2}} \quad (11.14)
 \end{aligned}$$

As the division term at the end is only slightly smaller and very close to 1, this expression is usually very close to just

$$G_M \approx -g_m \quad (11.15)$$

For example if we again assume that the transistors are biased to obtain a decent intrinsic gain $g_m r_{dsn} > 10$ and that all resistors $r_{dsn} = r_{dsp} = R_B$ are the same, as well as the transconductances $g_{mn} = g_{mp}$ we get:

$$-g_m < G_M < -g_m * \frac{200 + 40 + 2}{200 + 60 + 5} = -g_m * 0.91 \quad (11.16)$$

So for practical purposes we will usually just use (11.15).

And then R_O . From figure 11.10:

$$\begin{aligned}
 v_d \left(\frac{1}{r_{dsp}} + \frac{1}{r_{dsn}} + g_{mp} \right) &= v_s \left(\frac{1}{r_{dsn}} + g_{mn} \right) \\
 v_s \left(2g_{mn} + 2\frac{1}{r_{dsn}} + \frac{1}{R_B} \right) &= v_d \frac{1}{r_{dsn}} + v_o \frac{1}{r_{dsn}} \\
 v_o \left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} \right) + v_d g_{mp} &= v_s \left(\frac{1}{r_{dsn}} + g_{mn} \right) + i_o \quad (11.17)
 \end{aligned}$$

Solving this yields:

$$R_O = \frac{v_o}{i_o} = (r_{dsn} || r_{dsp}) \frac{2g_{mp}g_{mn} + 2\frac{g_{mn}}{r_{dsp}} + \frac{g_{mp}}{R_B} + 2\left(\frac{g_{mp}}{r_{dsn}} + \frac{1}{r_{dsn}r_{dsp}} + \frac{1}{r_{dsp}2R_B} + \frac{1}{r_{dsn}2R_B}\right) + \frac{g_{mn}}{r_{dsn}} + \frac{1}{r_{dsn}^2}}{2g_{mp}g_{mn} + 2\frac{g_{mn}}{r_{dsp}} + \frac{g_{mp}}{R_B} + 2\left(\frac{g_{mp}}{r_{dsn}} + \frac{1}{r_{dsn}r_{dsp}} + \frac{1}{r_{dsp}2R_B} + \frac{1}{r_{dsn}2R_B}\right)} \quad (11.18)$$

Once more the ratio at the end is an expression very close to 1, this time slightly bigger.

$$R_O \approx (r_{dsn} || r_{dsp}) \quad (11.19)$$

Again with the assumption of a decent intrinsic gain of $g_m r_{dsn} > 10$ and all resistors $r_{dsn} = r_{dsp} = R_B$ and transconductances $g_{mn} = g_{mp}$ being the same we get:

$$(r_{dsn} || r_{dsp}) < R_O < \frac{200 + 60 + 3}{200 + 50 + 2} (r_{dsn} || r_{dsp}) = 1.04 (r_{dsn} || r_{dsp}) \quad (11.20)$$

So for any practical purpose we will usually just use (11.19)

Combining the exact results (11.14) and (11.18) we confirm the result in (11.11) and get the same expression:

$$A = -G_M * R_O = g_m (r_{dsn} || r_{dsp}) * \frac{2g_{mn}g_{mp} + 2\frac{g_{mp}}{r_{dsn}} + \frac{g_{mp}}{R_B} + \left(\frac{g_{mn}}{r_{dsp}} + \frac{1}{r_{dsn}r_{dsp}} + \frac{1}{r_{dsn}2R_B} + \frac{1}{r_{dsp}2R_B}\right)}{2g_{mp}g_{mn} + 2\frac{g_{mn}}{r_{dsp}} + \frac{g_{mp}}{R_B} + 2\left(\frac{g_{mp}}{r_{dsp}} + \frac{1}{r_{dsn}r_{dsp}} + \frac{1}{r_{dsp}2R_B} + \frac{1}{r_{dsn}2R_B}\right)} \quad (11.21)$$

Again an expression close to:

$$A = -G_M R_O \approx g_m (r_{dsn} || r_{dsp}) \quad (11.22)$$

And once more for for example $g_m r_{dsn} > 10$ and all resistors $r_{dsn} = r_{dsp} = R_B$ and transconductances $g_{mn} = g_{mp}$ being the same we get:

$$g_m (r_{dsn} || r_{dsp}) * 0.953 < A < g_m (r_{dsn} || r_{dsp}) \quad (11.23)$$

11.3.3 Common mode rejection ratio (CMRR)

With the current mirror load the circuit is no longer exactly symmetrical. Thus, it can be expected that a change in the common mode input voltage $\hat{v}_i$, while the input difference $\Delta v_i = 0$, may actually cause the output to move (i.e. *also* already in the small signal world!!!). So if one does a complete analysis of the small signal equivalent circuit once again, but now with $\hat{v}_i$ as the input, one starts from the small signal model in figure 11.11.

The Kirchhoff current equations for the nodes are:

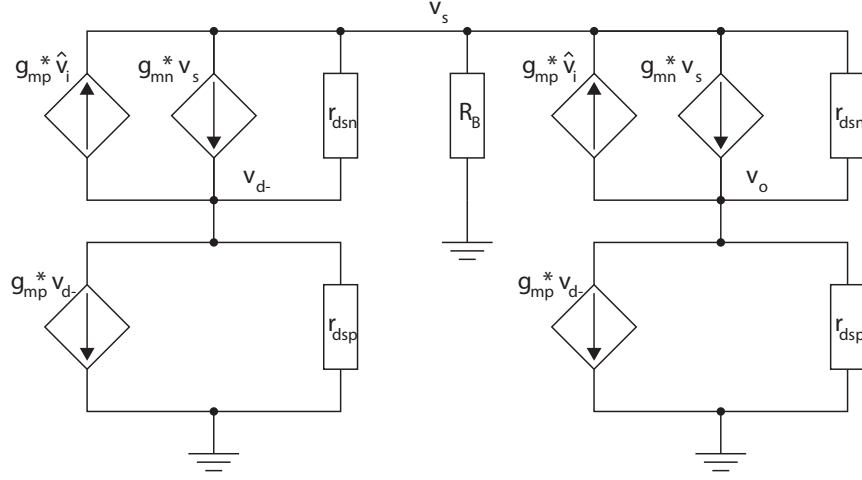


Figure 11.11: The small signal equivalent model for a differential pair with a current mirror load with common mode input $\hat{v}_i$.

$$\begin{aligned}
 v_d \left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} + g_{mp} \right) + \hat{v}_i g_{mn} &= v_s \left(g_{mn} + \frac{1}{r_{dsn}} \right) \\
 v_s \left(2 \frac{1}{r_{dsn}} + 2g_{mn} + \frac{1}{R_B} \right) &= v_d \frac{1}{r_{dsn}} + v_o \frac{1}{r_{dsn}} + 2\hat{v}_i g_{mn} \\
 v_o \left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} \right) + \hat{v}_i g_{mn} + v_d g_{mp} &= v_s \left(g_{mn} + \frac{1}{r_{dsn}} \right)
 \end{aligned} \tag{11.24}$$

The resulting common mode gain is:

$$\frac{v_o}{\hat{v}_i} = - \frac{\frac{g_{mn}}{R_B}}{2g_{mp}g_{mn} + 2\frac{g_{mp}}{r_{dsn}} + 2\frac{g_{mn}}{r_{dsp}} + \frac{g_{mp}}{R_B} + 2\frac{1}{r_{dsn}r_{dsp}} + \frac{1}{r_{dsn}R_B} + \frac{1}{r_{dsp}R_B}} \tag{11.25}$$

If we again make the assumption that the resistors r_{dsn}, r_{dsp} are equal and all transconductances g_{mn}, g_{mp} are equal and at least 10 times bigger than the inverted resistors (conductance of the resistors), then by only considering the term $2g_{mn}g_{mp}$ (which is at least 10 times bigger than any of the others) in the denominator one gets

$$\frac{v_o}{\hat{v}_i} \approx - \frac{1}{2R_B g_{mp}} \tag{11.26}$$

So if also R_B is approximately equal to the other drain source resistances, then the common mode gain is approximately equal to the inverse of (four times) the amplifiers differential gain.

The so called **common mode rejection ratio (CMRR)** is defined as the absolute ratio between the differential gain and the common mode gain:

$$\text{CMRR} := \left| \frac{\frac{v_o}{\Delta v_i}}{\frac{v_o}{\hat{v}_i}} \right| \tag{11.27}$$

If again we use the approximate terms we get:

$$\text{CMRR} \approx g_{mn} (r_{dsn} || r_{dsp}) 2g_{mp} R_B \quad (11.28)$$

11.3.4 Systematic offset

There is in fact a simple(r) intuitive way of deducing the common mode gain approximation in (11.26), and that is by looking at the expected output voltage when the differential input is zero ($\Delta v_i = 0$, i.e. we are still thinking small signal here). The first step is to understand that the differential pair with only a common mode input and zero differential input is in fact a source follower. Thus, the source voltage v_s of the diff pair transistors follows the common mode input v_{cm} :

$$v_s \approx v_{cm} \quad (11.29)$$

Consequently the bias current running through R_B is changing proportionally with v_{cm} and half of that goes through each branch of the differential pair. Let's call that i_d :

$$i_d \approx \frac{v_{cm}}{2R_B} \quad (11.30)$$

Now, using the fact that the current through the current mirror on top is given as:

$$\begin{aligned} i_d &\approx v_d g_{mp} \\ \Rightarrow \\ v_d &\approx \frac{i_d}{g_{mp}} \approx \frac{v_{cm}}{2R_B g_{mp}} \end{aligned} \quad (11.31)$$

where v_d is the drain voltage of the differential pair transistor at the *input* of the current mirror (neglecting r_{dsp} of the current mirror pFET).

Now an **important realization** is that if the differential input is zero, the drain voltage of the *output* of the current mirror will have to be the same voltage as on the input. That is because of symmetry reasons, i.e. all of the voltages and currents on the left branch other than v_d are given to be the same as on the right branch. That is only possible if also $v_d = v_o$ on the right settles on the same voltage as on the left. Thus we have again the same result as (11.26):

$$\frac{v_o}{v_{cm}} \approx \frac{1}{2R_B g_{mp}} \quad (11.32)$$

The realization above also carries to the large signal world, and from that we can make a useful deduction with respect to the output point of operation V_O . And that is that if we define the point of operation where the differential *input* is zero ($\Delta V_I = 0$) then the output point of operation will be one appropriate V_{SG} below Vdd such that the current mirror conducts $\frac{I_B}{2}$. It is thus important to note that this point of operation V_O is very unlikely to be at $\frac{V_{dd}}{2}$, and is usually closer to Vdd. In other words if one would *like* the output zero reference to be $\frac{V_{dd}}{2}$ then this deviation of the output from that value is a **systematic offset error**.

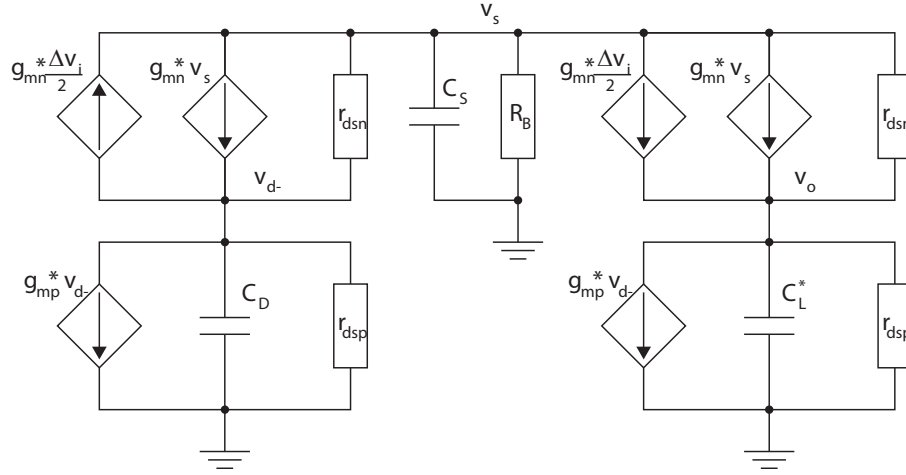


Figure 11.12: The small signal equivalent model for a differential pair with a current mirror load Including prominent capacitors for frequency analysis.

One can also regard this as an error at the input, i.e. the input differential voltage needs an offset from zero for the output to be the desired zero reference. So if the output point of operation is *defined* as $V_O := \frac{V_{dd}}{2}$, then the differential input suffers from an offset error.

Note that in addition to this *systematic* offset that assumes a symmetric design of the amplifier, there will be further offset caused by mismatching properties of the transistors in production.

11.3.5 Frequency response

Figure 11.12 shows the small signal model of the current mirror loaded differential pair, now including the most prominent capacitances. Note that some capacitors are neglected. More precisely some terms as specified below are neglected in the following Kirchhoff equations. The capacitors that *are* included are:

$$C_S = 2(C_{GS_n} + C_{SB_n}) + C_{DB_n} + C_{GD_n} \quad (11.33)$$

$$C_D = 2C_{GS_p} + C_{GD_p} + C_{DB_p} + C_{GD_n} + C_{DB_n} \quad (11.34)$$

$$C_L^* = C_{DB_n} + C_{GD_n} + C_{DB_p} + C_{GD_p} + C_{L_{ext}} \quad (11.35)$$

$C_{L_{ext}}$ is a contribution from the actual load the circuit is connected to, so often of the order of magnitude of C_{GS} , if that circuit is another CMOS circuit of similar dimensions.

The capacitors that are omitted in figure 11.12 and thus the missing terms in the Kirchhoff equations are:

- forward effect of C_{GD_n} between positive input $\frac{\Delta v_i}{2}$ and v_{d-} : the term $sC_{DG_n} \frac{\Delta v_i}{2}$ will be omitted in the following, assuming $C_D \gg C_{DG_n}$ it will not affect the voltage v_{d-} significantly.

- feedback effect of C_{GD_p} between v_{d-} and v_o : that is, we shall neglect the term $v_o C_{GD_p}$. This is a bit more daring, as there is some Miller effect to be considered here: v_{d-} is inverted from v_o and v_o is very much amplified. However, this amplification is likely small at the high frequencies where a pole at v_{d-} occurs (see in the following).
- The forward effect of C_{GD_n} and C_{GD_p} on node v_o . This is likely insignificant with external contribution to C_L^* by the circuit this amplifier is connected to.

We solve the problem here for ideal input voltage input signals. If, on the other hand, one wanted to consider the signal sources having some output impedance, one can as a first approximation consider the input nodes v_{i+} to have a static input capacitance of $C_{GS_n} + C_{GD_n}$, while the input node v_{i-} actually suffers from the miller effect and at low frequencies sees $C_{GS_n} + (1 + A)C_{GD_n}$. But once again: we do not do that here in the following, but consider ideal inputs.

Thus the resulting Kirchhoff equations are:

$$\begin{aligned}
 v_d \left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} + g_{mp} + sC_D \right) + \frac{\Delta v_i}{2} g_{mn} &= v_s \left(g_{mn} + \frac{1}{r_{dsn}} \right) \\
 v_s \left(2 \frac{1}{r_{dsn}} + 2g_{mn} + \frac{1}{R_B} + sC_S \right) &= v_d \frac{1}{r_{dsn}} + v_o \frac{1}{r_{dsn}} \\
 v_o \left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} + sC_L^* \right) + v_d g_{mp} &= v_s \left(g_{mn} + \frac{1}{r_{dsn}} \right) + \frac{\Delta v_i}{2} g_{mn}
 \end{aligned} \tag{11.36}$$

But lets go for G_M and Z_O separately, Z_O being what we previously called R_O but which is now an impedance rather than just a resistance.

The Kirchhoff equations to solve for G_M with the small signal output shorted to Gnd are:

$$\begin{aligned}
 v_d \left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} + g_{mp} + sC_D \right) + \frac{\Delta v_i}{2} g_{mn} &= v_s \left(g_{mn} + \frac{1}{r_{dsn}} \right) \\
 v_s \left(2 \frac{1}{r_{dsn}} + 2g_{mn} + \frac{1}{R_B} + sC_S \right) &= v_d \frac{1}{r_{dsn}} \\
 v_d g_{mp} = i_o + v_s \left(g_{mn} + \frac{1}{r_{dsn}} \right) + \frac{\Delta v_i}{2} g_{mn}
 \end{aligned} \tag{11.37}$$

Resulting in:

$$\frac{i_o}{\Delta v_i} = G_M = -g_{mn} \frac{\left(g_{mn} + \frac{1}{r_{dsn}} + \frac{1}{2R_B} + \frac{sC_S}{2} \right) \left(2g_{mp} + \frac{1}{r_{dsp}} + \frac{1}{r_{dsn}} + sC_D \right) - \frac{1}{r_{dsn}} \left(\frac{1}{r_{dsn}} + g_{mn} \right)}{\left(2g_{mn} + 2 \frac{1}{r_{dsn}} + \frac{1}{R_B} + sC_S \right) \left(g_{mp} + \frac{1}{r_{dsp}} + \frac{1}{r_{dsn}} + sC_D \right) - \frac{1}{r_{dsn}} \left(\frac{1}{r_{dsn}} + g_{mn} \right)} \tag{11.38}$$

And doing the Kirchhoff equations for Z_O with the small signal input being constant, i.e. at 0V:

$$\begin{aligned}
v_d\left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} + g_{mp} + sC_D\right) &= v_s\left(g_{mn} + \frac{1}{r_{dsn}}\right) \\
v_s\left(2\frac{1}{r_{dsn}} + 2g_{mn} + \frac{1}{R_B} + sC_S\right) &= v_d\frac{1}{r_{dsn}} + v_o\frac{1}{r_{dsn}} \\
v_o\left(\frac{1}{r_{dsn}} + \frac{1}{r_{dsp}} + sC_L^*\right) + v_dg_{mp} &= v_s\left(g_{mn} + \frac{1}{r_{dsn}}\right)
\end{aligned} \tag{11.39}$$

Solving for Z_O we get:

$$\begin{aligned}
\frac{v_o}{i_o} = Z_O &= (r_{dsn}||r_{dsp}) \frac{1}{1 + s(r_{dsn}||r_{dsp})C_L^*} \\
&\quad \left(\left(\frac{1}{r_{dsp}} + \frac{1}{r_{dsn}} + g_{mp} + sC_D \right) (2g_{mn} + 2\frac{1}{r_{dsn}} + \frac{1}{R_B} + sC_S) - \left(\frac{1}{r_{dsn}} + g_{mn} \right) \frac{1}{r_{dsn}} \right) \\
&\quad * \frac{1}{\left(\left(\frac{1}{r_{dsp}} + \frac{1}{r_{dsn}} + g_{mp} + sC_D \right) (2g_{mn} + 2\frac{1}{r_{dsn}} + \frac{1}{R_B} + sC_S) - 2\left(\frac{1}{r_{dsn}} + g_{mn} \right) \frac{1}{r_{dsn}} \right)}
\end{aligned} \tag{11.40}$$

Together this is:

$$\begin{aligned}
\frac{v_o}{\Delta v_i} = -G_M Z_O &= g_{mn}(r_{dsn}||r_{dsp}) \frac{1}{1 + s(r_{dsn}||r_{dsp})C_L^*} \\
&\quad \left(2g_{mp} + \frac{1}{r_{dsp}} + \frac{1}{r_{dsn}} + sC_D \right) \left(g_{mn} + \frac{1}{r_{dsn}} + \frac{1}{2R_B} + \frac{sC_S}{2} \right) - \left(g_{mn} + \frac{1}{r_{dsn}} \right) \frac{1}{r_{dsn}} \\
&\quad * \frac{1}{\left(g_{mp} + \frac{1}{r_{dsp}} + \frac{1}{r_{dsn}} + sC_D \right) \left(2g_{mn} + 2\frac{1}{r_{dsn}} + \frac{1}{R_B} + sC_S \right) - 2\left(g_{mn} + \frac{1}{r_{dsn}} \right) \frac{1}{r_{dsn}}}
\end{aligned} \tag{11.41}$$

Making the same assumptions again that $g_m r_{ds} \gg 1$ is orders of magnitude bigger than 1, we neglect everything except terms of the order g_m^2 or $sC_S g_m$ or sC^2 . We get:

$$\frac{v_o}{\Delta v_i} = -G_M Z_O \approx g_{mn}(r_{dsn}||r_{dsp}) \frac{1}{1 + s(r_{dsn}||r_{dsp})C_L^*} \frac{\left(1 + s\frac{C_D}{2g_{mp}}\right)}{\left(1 + s\frac{C_D}{g_{mp}}\right)} \tag{11.42}$$

So we have the pole frequency ω_{p_1} resulting from the output impedance Z_O :

$$\omega_{p_1} = \frac{1}{(r_{dsn}||r_{dsp})C_L^*} \tag{11.43}$$

It is quite easily the dominant pole, since the other pole frequency ω_{p_2} (resulting from the expression for G_M) is close to the maximum frequency one of the diff pair transistors can convey under ideal circumstances, the intrinsic speed, i.e. the current unity gain/transition frequency $\omega_T = \frac{g_m}{C_{GS} + C_{GD}}$ (see section 9.5.1, equation(9.39)) (here: $\omega_T = \frac{g_m}{C_L^*}$):

$$\omega_{p_2} = \frac{g_{mp}}{C_D} \tag{11.44}$$

Another interesting connection is that this pole frequency is identical to the one derived as the input related cut off frequency of a current mirror with the

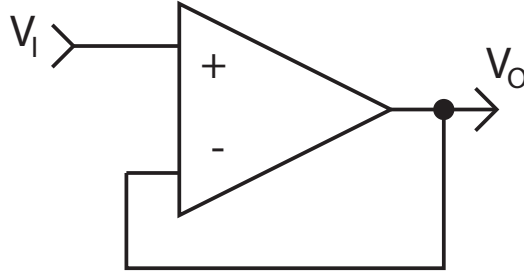


Figure 11.13: A voltage follower implemented with an operational amplifier in a feedback loop. the feedback factor $\beta = 1$ in this case and thus the closed loop gain becomes $A_{cl} \approx \frac{1}{\beta} = 1$.

output connected to ground in equation (10.25). So that is where the current mirror starts to shut down.

The sole zero frequency ω_{z_1} also results from the G_M term and occurs at double the frequency:

$$\omega_{z_1} = 2\omega_{p_2} = \frac{2g_{mp}}{C_D} \quad (11.45)$$

The combined effect of these two later terms is that the output current starts to fall proportional with the frequency with an 1:1 ratio or $20 \frac{\text{dB}}{\text{dec}}$ at ω_{p_2} and stops at double that frequency at $\omega_{z_1} = 2\omega_{p_2}$. Consequently, $|G_M|$ and thus the output current is *halved* at that point. The intuition here is that the current mirror goes into cut off, so only the output current from the right diff pair branch drives the voltage gain. But once again: this usually happens at way higher frequencies than ω_{p_1} , i.e. beyond the circuit's unity gain frequency. So oftentimes this amplifier is modelled as a simple first order system with the only pole being p_1 .

11.3.6 Slew rate

The **slew rate** is defined as a maximum rate of change of output voltage. It is a phenomenon that can lead to significant distortion and signal attenuation *within the pass band* of an amplifier. It is basically the large signal behaviour nipping the small signal model in the but. It's best understood by looking at the large signal current output of a transconductance amplifier. The linear range of that current output is limited by $\pm I_B$. So as long as the input difference is small enough (i.e. within the input linear range $\Delta V_I \leq \Delta V_{I_{max}}$ (in strong inversion that is $\Delta V_I \leq \sqrt{2}V_{OV}$) around the point of operation V_I), the output current will be linearly dependent, like the small signal model assumes and predicts. However, if the input difference falls outside the linear range of the transconductance amplifier, the small signal assumption falls apart and the current becomes constant either I_B or $-I_B$.

A sine wave signal of a frequency below the cut off frequency of the amplifier will appear as an undistorted, amplified sine wave voltage signal at the output as long as the input amplitude is small (see top figure 11.14). The small signal

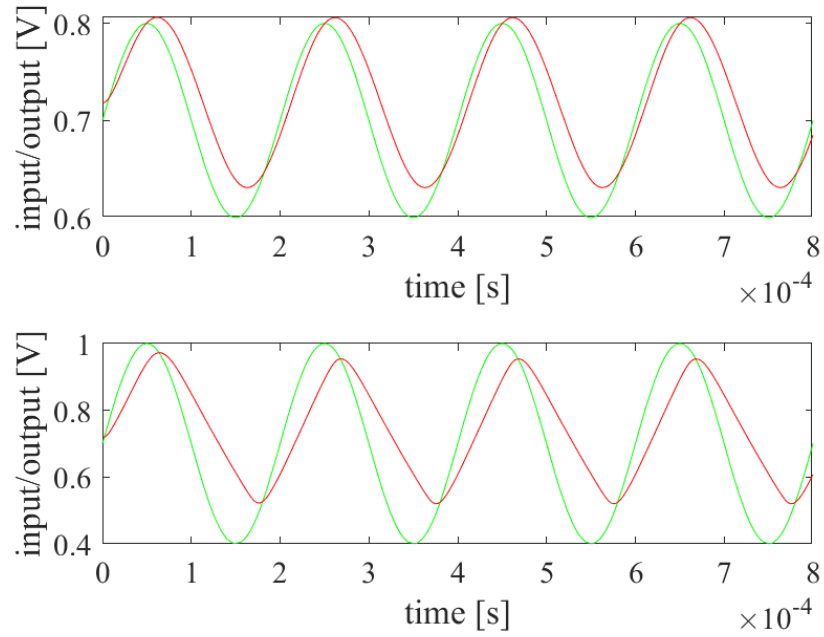


Figure 11.14: An illustration of the effect the slew rate can have. It is the result of a cadence simulation with a transconductance amplifier connected as a follower (the output shorted to the negative input terminal) with a load capacitor. The output (red) follows the input (green) quite faithfully in this example if the input has an amplitude of 100mV at a frequency of 5kHz. The output amplitude is not fully as large as the input and experiences a slight phase shift, due to the not infinite gain of the transconductance amplifier and because the cut off frequency starts to have an effect, even though it's higher than 5kHz. Still the output is a sine wave. If we increase the amplitude of the input to 300mV, the output becomes distorted and attenuated due to the slew rate.

model is thus an appropriate description of what happens. However, if the input signal is bigger than the input linear range, the output waveform becomes a distorted sine wave wherever the expected small signal rate of change $\frac{\Delta v_i g_m}{C_L^*} \left| \frac{\partial v_o}{\partial t} \right|$ exceeds $\left| \frac{I_B}{C_L^*} \right|$ with an attenuated amplitude (see bottom figure 11.14).

This is not very likely to be the first concern in open loop transconductance amplifiers because of the high gain: for large input amplitudes the voltage supply limits will become a problem for the output before the slew rate does. On the other hand, the slew rate can easily become a problem when negative feedback is used to obtain a reduced closed loop gain, most severely in the case of a follower circuit (compare figure 11.13) where the gain is reduced to 1, i.e. 0dB (as has been done to obtain figure 11.14). In these closed loop configurations the amplifier attempts via the loop to maintain a zero difference at the input terminals. However, if that would require the output to move faster than the slew rate allows, the input terminal difference will exceed the input linear range and the output will now move in a straight line only as fast as I_B allows.

For commercial amplifiers the spec sheet will indicate slew rate as the maximum rate of change of the output voltage, and will not depend on the load capacitance. That is because commercial amplifiers are usually multi stage amplifiers internally and the origin of the slew rate will be at an internal node, i.e. not the output node. A common configuration is a two stage amplifier, where the first stage is a transconductance amplifier that is subject to such a maximum current, but then also an output stage that is a much better voltage source, with very small output resistance that can drive large loads. The slew rate will then occur at the interconnection between those two stages caused by the maximum current in that node and will manifest as a maximum rate of change of output voltage (rather than a maximum current at the output node).

11.4 Higher gain operational differential amplifiers introduction

In the following there are two amplifier configurations mentioned briefly that achieve about quadratically higher gain than the single stage differential transconductance amplifier, with only a partial analysis. The term *operational* is added to really high gain amplifiers and historically stems from their original use in performing arithmetic operations in analog computers.

The examples here are still transconductance amplifiers, meaning they are better suited as providing stable current outputs. Operational voltage amplifiers will have another (usually unity gain) output stage to provide a low output impedance and thus rendering the output more suited as voltage source that can drive low resistance loads.

11.4.1 Folded cascode operational transconductance amplifier

Figure 11.15 shows the popular folded cascode operational transconductance amplifier. It is to the transconductance amplifier what the folded cascode gain stage is to the CS gain stage.

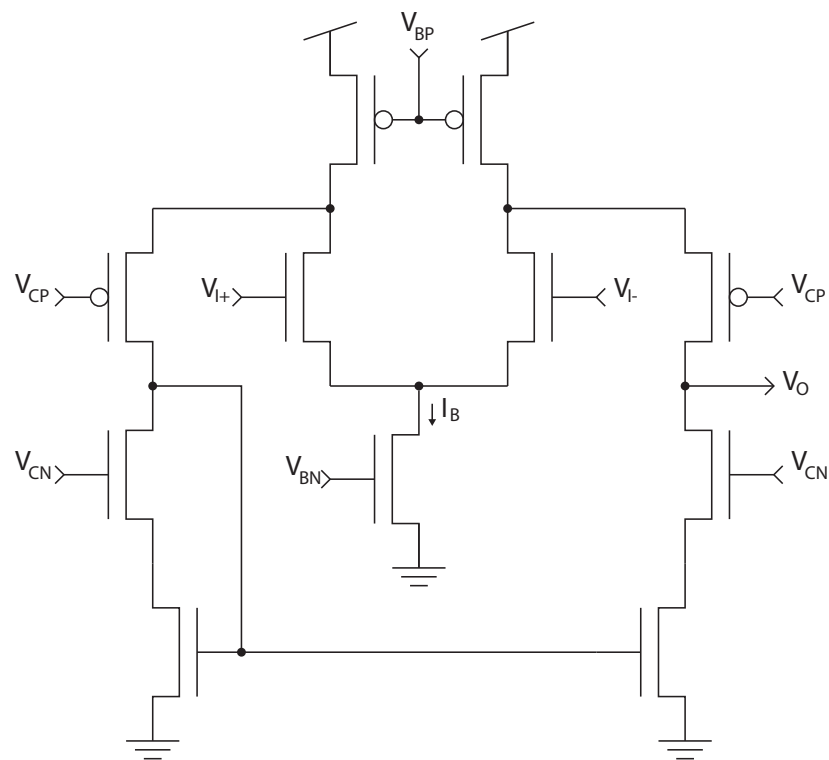


Figure 11.15: Differential input, single ended output folded cascode operational transconductance amplifier

So we shall not go into the intricacies of paper and pen analysis of it, but name a few properties. It is to the transconductance amplifier (current mirror loaded differential pair) what the folded cascode gain stage is to the CS gain stage: The differential gain in open loop is approximately the square of the gain of the transamp. The Miller effect on the input is reduced (that's in fact only the negative input of the transamp that suffers from this) and the pole on the input node is shifted to a higher frequency. The pole at the output node is subject to the much increased output resistance, so more likely to be the dominant one. With that heightened output resistance it is in fact imperative to have an additional output stage with low output resistance (not discussed in this compendium) if this circuit is to drive any noticeable capacitive or small resistive load. Take care when tuning the bias voltages V_{CP} and V_{CN} for the cascode transistors. Check the musings on cascode current mirrors and equation (10.34) for some pointers. Also note that the saturation currents I'_D in the bias transistors with V_{BP} at their gate need not just balance I_B but provide a surplus to bias the cascode branches with some non-zero current. Note that the choice of these voltages also influences the linear output range which will be diminished as compared to the simple differential transamp.

11.4.2 Two stage operational transconductance amplifier

An alternative choice is the simple differential transamp with a second common source stage (figure 11.16). It offers similar gain to the folded cascode amplifier (which is in essence a two stage amplifier too). It has a wider output range and a more forgiving biasing range with only one bias voltage, but potentially a bigger offset and generates a pole of lower frequency at its internal node between the stages due to the Miller effect there. This Miller effect however, can also be exploited and increased by an additional *explicit* feedback capacitance between V_1 and V_O to move said pole to an even lower frequency to stabilize a feedback loop that uses this amplifier (see 'pole separation'). This is referred to as **compensation**, but is a topic that we will not cover in this compendium.

The ratio of the differential pair current mirror pFETs' aspect ratio $\frac{W_{p1}}{L_{p1}}$ to the CS' pFETs $\frac{W_{p2}}{L_{p2}}$ should be close to half the ratio of the respective bias nFETs' $\frac{W_{n1}}{L_{n1}}$ and $\frac{W_{n2}}{L_{n2}}$:

$$\frac{W_{p1}L_{p2}}{L_{p1}W_{p2}} \approx \frac{W_{n1}L_{n2}}{2L_{n1}W_{n2}} \quad (11.46)$$

It is only *approximate* due to the channel length modulation (i.e. if $\lambda \neq 0$) and the fact that the drain voltages of the bias nFETs are different. Without channel length modulation (i.e. if $\lambda = 0$), if (11.46) were exact, we would have $\Delta V_I = 0 \Rightarrow V_O = V_1$. So likely the same offset as the one stage differential transconductance amplifier, i.e. a voltage somewhat closer to Vdd than Gnd, i.e. not $\frac{V_{dd}}{2}$. The channel length modulation and the difference in drain voltage for the bias nFETs in the two stages is partially counteracting that shift, as the differential pair bias current will be slightly smaller than double the CS bias current. Furthermore, also mismatch between designed aspect ratios and produced aspect ratio will have an additional random influence on the offset, but that's the case for any amplifier and system design needs to take that into account.

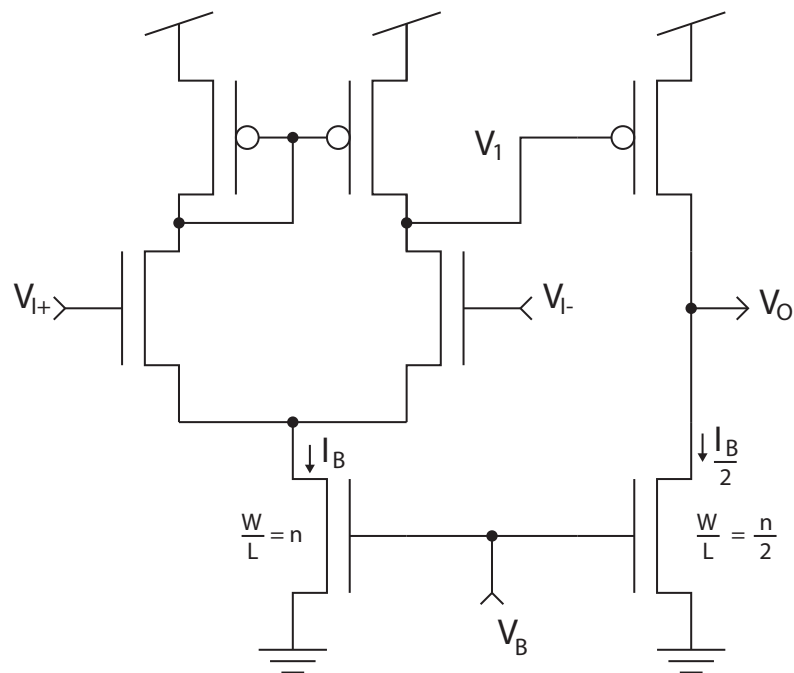


Figure 11.16: Differential input, single ended output two stage operational amplifier. Making the nFET of the second stage half as wide as the bias transistor of the differential pair is but one approximate solution to keep the input referred offset small.

Index

- C_L^* , 127
- C_I , 90
- C_O , 90
- C_W , 90
- C_{ox} , 10
- I_D' , 27
- I_S , 10
- L , 10
- R_I^* , 127
- R_L^* , 127
- R_S , 56
- U_T , 10
- V_{eff} , 11
- V_t , 10
- V_{OV} , 11
- W , 10
- Z_S , 56
- λ , 10
- μ , 10
- ω_H , 62
- ω_T , 131
- g_m , 26, 27
- g_{mb} , 26
- n , 10
- r_{ds} , 26, 27
- t_p , 35, 40
- 3dB, 62

- above threshold region, 18
- accumulation, 4
- active region, 18
- admittance, 52
- asynchronous, 105, 109

- B, 3
- back gate, 3, 4, 11
- back gate effect, 26, 28
- back gate transconductance, 26
- bias generator, 139
- biasing, 21, 120
- binary, 35

- Bode plot, 56
- Boolean functions, 43
- BOX, 4
- bulk, 3
- bulk CMOS, 3, 4
- buried oxide, 4

- carrier mobility, 10
- cascode, 124
- CD, 122
- CG, 122
- channel length, 10
- channel length modulation parameter, 10
- channel width, 10
- characteristic current, 10
- characteristic frequency, 64
- characteristic table, 105
- CLK, 108
- clock, 108
- closed loop gain, 71
- closed loop gain error, 73
- CMOS, 3
- CMRR, 156, 163, 164
- combinational logic circuits, 43, 85
- Common Drain, 122
- Common Gate, 122
- common mode input, 155
- common mode input voltage, 156
- common mode rejection ratio, 156, 163, 164
- Common Source, 120
- comparator, 155
- compensation, 173
- complementary metal oxide semiconductor, 3
- complementary pull-up and pull down networks, 85
- complementary pull-up/pull-down, 88
- conduction, 11, 16, 18
- cross current, 94

- CS, 120
- current conveyer, 122
- current mirror, 137
- cut off frequency, 62
- D, 3
- D-latch, 105
- design parameters, 10
- differential amplifiers, 153
- differential input, 155
- differential transconductance amplifier, 153
- diode connected FET, 138
- dominant pole, 62
- domino logic, 100
- drain, 3
- drain resistance, 26, 27
- dual rail Domino logic, 100
- dynamic logic, 88, 89, 97
- dynamic power consumption, 94
- effective voltage, 11
- EKV model, 9
- evaluation phase, 97
- FET, 3
- field effect transistor, 3
- FinFET, 5, 7
- finite state machine, 110, 111
- finite state machines, 43
- Fourier Transform, 48
- FSM, 43, 110, 111
- function, 48
- G, 3
- gain bandwidth product, 132
- gain error, 73
- gate, 3
- gating, 108
- GBW, 132
- impedance, 47, 52
- input impedance, 55
- input impedance/resistance, 55
- input voltage range, 28
- intrinsic gain, 118, 121
- inversion, 4
- Kirchhoff's current law, 47
- Kirchhoff's voltage law, 47
- large signal, 21
- length, 10
- linear filter, 47
- linear region, 18
- linear system, 47
- logic gates, 85, 86
- loop gain, 72
- low frequency small signal equivalent, 24
- maximally flat response, 66
- Mealy finite state machine, 111
- metastable state, 105
- moderate inversion, 16
- Moore finite state machine, 111
- MOSFET, 3
- n-channel field effect transistor, 3
- nFET, 3
- Norton equivalent circuit, 55
- NP domino logic, 100
- open loop gain, 71
- operational amplifiers, 171
- operator, 48
- output impedance, 55, 56
- output impedance/resistance, 55
- output resistance, 56
- overdrive voltage, 11
- oxide capacitance, 10
- p-channel field effect transistor, 3
- pFET, 3
- phase margin, 74, 75, 77
- PM, 74, 75, 77
- PO, 27
- point of operation, 21, 27, 120
- pole frequency, 58
- pole separation, 73
- poles, 58
- precharge phase, 97
- process parameters, 10
- propagation delay, 35, 40
- pull-down, 35, 88
- pull-up, 35, 88
- Q-factor, 64
- quality factor, 64
- race conditions, 109
- resistance, 47

reversed biased junction, 4
roll off, 62
root form, 58

S, 3
saturation, 11, 16, 17
saturation current, 27
sequential logic, 105
sequential logic circuits, 43, 105
SF, 122
silicon on insulator, 3, 4
SiO, 3
slew rate, 169, 171
slope factor, 10
small signal, 21
small signal equivalent, 24, 29
SOI, 4
source, 3
Source Follower, 122
staggered clock, 98
state transition graph, 109
static power consumption, 94
strong inversion, 11, 16–18
subthreshold region, 18
supply rails, 9
switch, 35
synchronous, 108, 110
systematic offset, 165

thermal voltage, 10
threshold voltage, 10
Thévenin equivalent circuit, 55
transconductance, 26, 27
transconductance amplifier, 158
transfer function, 49
transition frequency, 131
transmission gate, 35, 36
transparent latch, 105
triode region, 18
truth table, 83

unity gain frequency, 63, 77, 131

weak inversion, 11, 16–18
width, 10

zero frequency, 58
zeros, 58

Bibliography

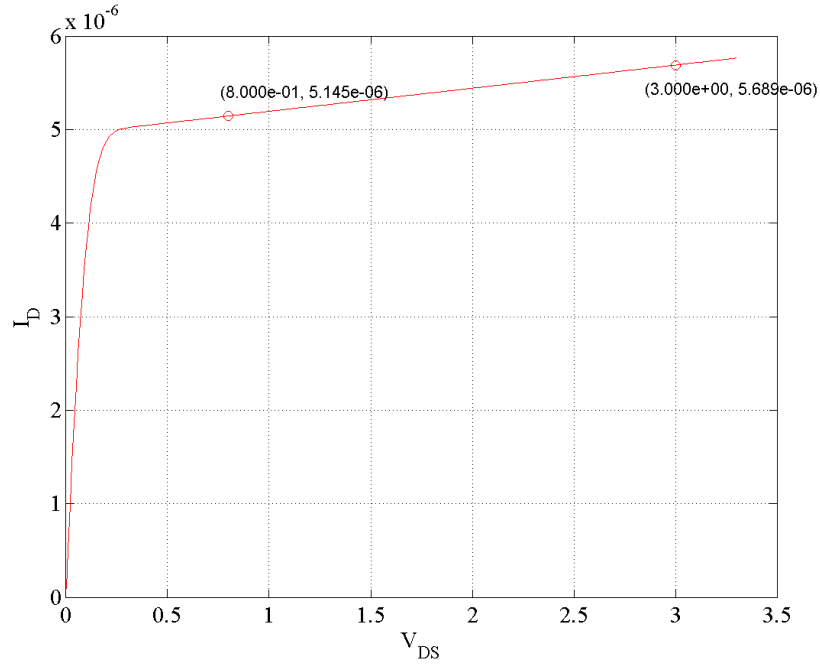
- [1] T. C. Carusone, D. Johns, and K. Martin. *Analog Integrated Circuit Design, international student version*. Wiley, 2013.
- [2] A. S. Sedra and K. C. Smith. *Microelectronic Circuits*. Oxford University Press, international 7th edition, 2016.
- [3] E. Vittoz. Analog VLSI signal processing: Why, where and how? *Analog Integrated Circuits and Signal Processing*, pages 27–44, July 1994.

Part V

Appendices

Appendix A

Exercises

Figure A.1: Exercise: compute λ from I_D vs V_{DS} graph

A.1 Transistor Models

A.1.1

From the I_D vs V_{DS} graph in figure A.1 for a specific nFET with a specific V_{GS} , compute parameter λ .

A.1.2

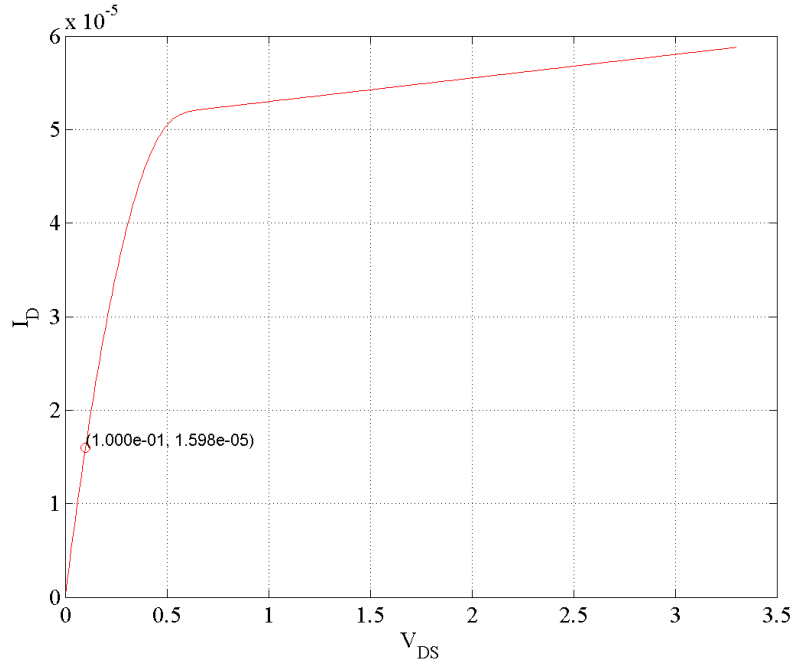
From I_D vs V_{DS} graph in figure A.2 for a specific nFET with a specific V_{GS} , compute parameter r_{ds} near the origin, i.e. in deep conduction.

A.1.3

An nFET is to be operated in the saturation region at a current of $20\mu A$. It's $k'_n \frac{W}{L} = 70 \frac{\mu A}{V^2}$ and its threshold voltage $V_{tn} = 650mV$. (You may assume parameter $\lambda = 0$ and $n = 1$). What is its required gate to source voltage $V_{GS} = ?$ and minimum drain to source voltage $V_{DS,min} = V_{sat} = ?$

A.1.4

Make a matlab function 'function '[Vd]=CMOSdiode(Id,kn,Vtn)' that computes the drain voltage Vd (V_D) on a diode connected nMOSFET (i.e. drain and gate are shorted together) as a function of the input current Id (I_D) at the drain in weak inversion (!!!), and given parameters kn, Vtn (i.e. the MOSFET

Figure A.2: Exercise: compute r_{ds} near origin from I_D vs V_{DS} graph

transconductance parameter k_n and the threshold voltage V_{tn}). Ignore channel length modulation $\lambda = 0$ and slope factor $n = 1$. The thermal voltage U_T shall be 26mV. Assume the source voltage $V_S = V_B = 0$.

Also assume that the transistor is in saturation. Could you find out from the result, if that is actually a correct assumption?

A.1.5

An NMOS transistor is operated in the triode region with drain source voltage close to zero $V_{DS} \approx 0$. Find the expressions for its drain resistance r_{ds} in dependency of k_n and overdrive voltage V_{OV} for both strong AND weak inversion. (If relevant, assume the slope factor $n = 1$.) Implement these two expressions as a MATLAB function with the header 'function [r_ds]=drainresistance(V_ov,k_n)' in file 'drainresistance.m'. Distinguish the strong and weak inversion case with:

```
if V_ov>0
    r_ds= <your strong inversion expression>;
else
    r_ds= <your weak inversion expression>;
end;
```

A.1.6

A nMOS and a pMOS of the same dimensions in the same technology have different carrier mobilities and thus $k_n = 2.2 * k_p$. If they both are operated in

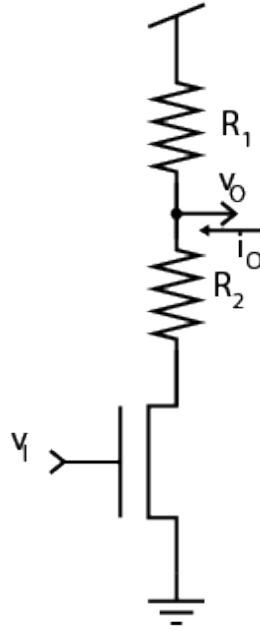


Figure A.3: Schematic for small signal analysis

saturation mode and in strong inversion, what must the overdrive voltage of the pMOS V_{OVp} be as a function of the overdrive voltage V_{OVn} of the nMOS if they both should conduct the same current? (Ignore any channel length modulation effects or slope factor, i.e. $\lambda = 0$ and $n = 1$!)

Submit a MATLAB function with header 'function [V_OVp]=pMOSOverdrive(V_OVn)', in file 'pMOSOverdrive.m' that computes V_{OVp} from V_{OVn} .

A.1.7

For the schematics in figure A.3 perform a small signal (!) analysis and find an expression for the total output resistance $R_O = \frac{v_o}{i_o}$ and the total transconductance $G_M = \frac{i_o}{v_i}$ and the gain $A = \frac{v_o}{v_i}$! To form your expressions you may use the small signal variables g_m, r_{ds} of the transistor as well as the resistors R_1, R_2 . In other words express these terms as functions of those four variables (or a subset of them).

A.1.8

Given process parameters: $\mu_n C_{ox} = 280 \cdot 10^{-6} \frac{\text{A}}{\text{V}^2}$, $\frac{1}{\lambda \cdot L} = \frac{V_A}{L} = 5 \cdot 10^6 \frac{\text{V}}{\text{m}}$, $V_{tn} = 0.5\text{V}$, for an NFET of length $L = 0.4\mu\text{m}$, $V_{ov} = 0.15\text{V}$, and operated at a drain current $I_D = 80\mu\text{A}$, find transconductance, drain resistance in the active/saturation region, intrinsic gain and transistor width: g_m, r_{ds}, A_0, W !

A.1.9

Make a matlab function 'function '[Vd]=CMOSdiode(Id,kn,Vtn)' that computes the drain voltage Vd (i.e. V_D in the compendium) on a diode connected nMOS-FET (i.e. drain and gate are shorted together) as a function of the input current Id (i.e. I_D in the compendium) at the drain in weak inversion (!!!), and given parameters kn (i.e. the MOSFET transconduction parameter k_n), Vtn (i.e. the threshold voltage V_{tn}) . Ignore λ (i.e. $\lambda = 0$) and V_A (the Early effect, i.e. $V_A = \infty$) and n (the slope factor from the EKV model) and you may assume that the transistor is in saturation. The thermal voltage U_T shall be 26mV. Assume the source voltage $V_S = 0$ and that all voltages are referenced to the bulk $V_B := 0$.

A.1.10

A nFET at room temperature (i.e. $U_T = 26\text{mV}$) has the following process- and design parameters:

$$\mu C_{ox} = 200 \frac{\mu\text{A}}{\text{V}^2}$$

$$\frac{W}{L} = 3$$

$$n = 1.2$$

$$V_t = 0.4\text{V}$$

$$V_S = V_B = 0\text{V} \text{ (i.e. the source is at ground as well as the bulk)}$$

If you assume that it is operated in saturation and that $\lambda = 0$, for which of these drain currents I_D will it be in weak inversion? Select all alternatives:

- ☐ 200pA
- ☐ 2nA
- ☐ 20nA
- ☐ 200nA
- ☐ 2 μA
- ☐ 20 μA

If you assume that it is operated with a gate voltage of $V_G = 0.7\text{V}$, (and $\lambda > 0$) for which of these drain currents I_D can the transistor be in the conduction region of operation? Select all alternatives:

- ☐ 100pA
- ☐ 1nA
- ☐ 10nA
- ☐ 100nA
- ☐ 1 μA
- ☐ 10 μA

A.1.11

Assume you have an nFET with $V_S = \text{Gnd}$ $V_{GS} = \text{Vdd}$ and a pFET with $V_S = \text{Vdd}$ and $V_{SG} = \text{Vdd}$ (so the *difference* is Vdd, i.e. $V_G = 0\text{V}$) and they have their drain terminals shorted together. Both shall have $\frac{W}{L} = 1$. Using process parameters from table A.1 except for assuming $\lambda = 0$ and $n = 1$, and the resistive switch model with the appropriate 'holding' resistance R_{ON} , what will be the voltage at the drain for the 180nm parameters and the 45nm parameters?

A.1.12

What is the propagation delay for a pFET pulling a voltage up from Gnd to $\frac{\text{Vdd}}{2}$ and an nFET pulling a voltage down from Vdd to $\frac{\text{Vdd}}{2}$ for transistors with $\frac{W}{L} = 1$ and the process parameters from table A.1, with the exception of $n = 1$ and $\lambda = 0$. Assume a total load capacitance $C_L = 50\text{fF}$. How much does the result differ if you use the model for I_{ON} from the model for R_{ON} ? Compute the same numbers also if you reduce the supply Vdd to a voltage that is $\frac{1}{3}$ below the threshold voltage. If the threshold voltages of pFET and nFET are different, use the smaller of the two.

A.1.13

Use the process parameters below and an nFET with width $W = 2\mu\text{m}$ and length $L = 0.3\mu\text{m}$! What is its characteristic current (use IS as variable name in MATLAB) as used in the EKV model?

Furthermore, biasing it with large signal currents (ignoring channel length modulation effects, i.e. these are what the compendium calls I'_D) Id1=1e-9, Id2= 1e-7, Id3=1e-5, Id4= 1e-3, what are its intrinsic gains A01, A02, A03, A04 for each of those currents?

Use the appropriate strong or weak inversion equation in each case! Source and bulk are both shorted to Gnd. Use these parameters (always pay close attention to the units, e.g. milli-, micro-, etc.):

$$U_T = 23\text{mV} \quad (\text{A.1})$$

$$\mu C_{ox} = 300 \frac{\mu\text{A}}{\text{V}^2} \quad (\text{A.2})$$

$$V_t = 0.5\text{V} \quad (\text{A.3})$$

$$\text{Vdd} - \text{Vss} = 2.5\text{V} \quad (\text{A.4})$$

$$\lambda L = 0.05_{\mu}\text{mV} \quad (\text{A.5})$$

$$n = 1 \quad (\text{A.6})$$

$$W = 2\mu\text{m} \quad (\text{A.7})$$

$$L = 0.3\mu\text{m} \quad (\text{A.8})$$

$$(\text{A.9})$$

Upload a matlab script 'IntrinsicGain.m' that assigns the results to the variables

IS= ...

A01= ...

A02= ...
A03 = ...
A04= ...

Solutions section A.1

Solution A.1.1

r_{ds} is the inverse of the slope in the saturation region, i.e. the inverse of the small signal conductance g_{ds} and λ is simply the conductance divided by the saturation drain current I'_D neglecting λ . Since we do have two points of that slope we can deduce:

$$g_{ds} = \frac{I_{D2} - I_{D1}}{V_{DS2} - V_{DS1}} = (5.689e-6 - 5.145e-6)/(3.0 - 0.8) = 0.2473e-6 \quad (\text{A.10})$$

One could eyeball $I'_D \approx 5e-6$, but more precisely:

$$I'_D = I_{D1} - V_{DS1} * g_{ds} = 5.145e-6 - 0.8 * 0.2473e-6 = 4.95e-6 \quad (\text{A.11})$$

$$\begin{aligned} I_{D1} &= I'_D(1 + V_{DS1} * \lambda) \quad 1) \\ I_{D2} &= I'_D(1 + V_{DS2} * \lambda) \quad 2) \\ &\Rightarrow 2) - 1) \\ I_{D2} - I_{D1} &= I'_D(V_{DS2} - V_{DS1}) * \lambda \\ \Rightarrow \lambda &= \frac{I_{D2} - I_{D1}}{I'_D(V_{DS2} - V_{DS1})} = \frac{g_{ds}}{I'_D} \\ &= 0.2473e-6 / 4.95e-6 = 0.05 \frac{1}{V} \end{aligned} \quad (\text{A.12})$$

Solution A.1.2

As the point labelled on the curve is quite close to the origin one can simply approximate:

$$r_{ds} \approx \frac{V_{DS1}}{I_{D1}} = 0.1 / 15.98e-6 = 6.26k\Omega \quad (\text{A.13})$$

Which is slightly overestimated as the curve is marginally steeper right at the origin.

Solution A.1.3

Assuming strong inversion (needs to be verified in the end) we can write

$$V_{ov} = \sqrt{\frac{2I_D}{k_n}} = \sqrt{2 * 20e-6 / 70e-6} = 0.756V \quad (\text{A.14})$$

Checking for strong inversion: V_{ov} is of the same order of magnitude as V_t . If it would be orders of magnitude smaller, the result could be wrong! In that case we would likely be in weak inversion. However, here we seem to be safe.

$$V_{GS} = 1.406V$$

$$V_{sat} = 0.756V$$

Solution A.1.4

In saturation with $\lambda = 0$ the current does not depend on V_{DS} at all, so one simply needs to find the correct V_G .

```
function '[Vd]=CMOSdiode(Id,kn,Vtn)
VT=26e-3 ;%V
Is=2*kn*VT^2;
% Id=Is*exp((Vg-Vtn)/VT) % solve for Vg
% log(Id)=log(Is)+(Vg-Vtn)/VT
Vg=VT*(log(Id)-Log(Is))+Vtn;
```

A diode connected CMOS transistor in strong inversion (so not here) is except for very artificial scenarios actually guaranteed to be in saturation, as $V_{sat} < V_{GS}$. So by shorting $V_{DS} = V_{GS}$, V_{DS} will always be big enough. Here in weak inversion however, for very small input currents assuming saturation the resulting V_{GS} might turn out to be smaller than $4U_T$. Then one would actually need to solve the problem using the conduction rules or full EKV and the real $V_{GS} = V_{DS}$ will turn out to be bigger.

Solution A.1.5

```
function [rds]=drainresistance(V_ov,k_n)
if V_ov>0
    r_ds= 1/(k_n*V_ov);
else
    VT=26e-3;
    IS=2*k_n*VT^2;
    r_ds= 1/(IS*exp(V_ov/VT)/VT);
end;
```

Solution A.1.6

```
function [V_OVp]=pMOSOverdrive(V_OVn)
V_OVp=V_OVn*sqrt(2.2)
end;
```

Solution A.1.7

$$R_O = (r_{ds} + R_2) || R_1 = \frac{(r_{ds} + R_2)R_1}{R_1 + R_2 + r_{ds}}$$

$$G_M = g_m \frac{r_{ds}}{r_{ds} + R_2}$$

$$A = G_M R_O = g_m \frac{R_1 r_{ds}}{R_1 + R_2 + r_{ds}}$$

Solution A.1.8

With given positive V_{ov} away from moderate inversion we can safely assume strong inversion. Thus:

$$\begin{aligned}
 W &= \frac{2I_D L}{V_{ov}^2 \mu C_{ox}} = 2 * 80e-6 * 0.4e-6 / (0.15^2 * 280e-6) = 10.16 \mu\text{m} \\
 g_m &= \sqrt{2I_D k_n} = \sqrt{2 * 80e-6 * 280e-6 * 10.16e-6 / 0.4e-6} = 1.07 \frac{\text{mA}}{\text{V}} \\
 r_{ds} &= \frac{1}{I_D \lambda} = 1 / (80e-6 * 1/5e6 / 0.4e-6) = 25 \text{k}\Omega \\
 A_0 &= g_m r_{ds} = 1.07e-3 * 25e3 = 26.75 \frac{\text{V}}{\text{V}}
 \end{aligned} \tag{A.15}$$

Solution A.1.9

The assumption that we are in weak inversion and saturation allows to use the simplified equation for that region of operation. However, if the input current 'Id' is too small or too large in relation to the other parameters, these assumptions may actually turn out to be wrong: if 'Id' is too small the resulting 'Vd' may be smaller than $4U_T$ and thus the transistor is actually in conduction and the result is wrong. Or if 'Id' is too large, it might be bigger than I_S and thus the transistor would really be in strong inversion. So the following matlab program only returns the correct results for certain inputs. But since the tasks specifies that one can assume weak inversion and saturation, it is the correct answer.

```

function [Vd]=CMOSdiode(Id,kn,Vtn)
VT=26e-3 ;%V
Is=2*kn*VT^2;%A
% Id=Is*exp((Vg-Vtn)/VT) % solve for Vg
% log(Id)=log(Is)+(Vg-Vtn)/VT
Vg=VT*(log(Id)-Log(Is))+Vtn;

```

To write down the underlying symbolic solution:

$$\begin{aligned}
 I_S &= 2k_n U_T^2 \\
 V_G &= U_T (\ln I_D - \ln I_S) + V_{tn}
 \end{aligned} \tag{A.16}$$

Solution A.1.10

a)

The simplest solution is to use I_S as the current magnitude that separates weak from strong inversion in saturation. Use equation (2.5)

$$I_S = 2 * 1.2 * 3 * 200e-6 * (26e-3)^2 = 973 \text{nA}$$

the Compendium mentions two rules that are used in literature where either $I_S = 973 \text{nA}$ or $0.48 * I_S = 467 \text{nA}$ (the latter is equivalent to $V_{OV} = 0$) is used as

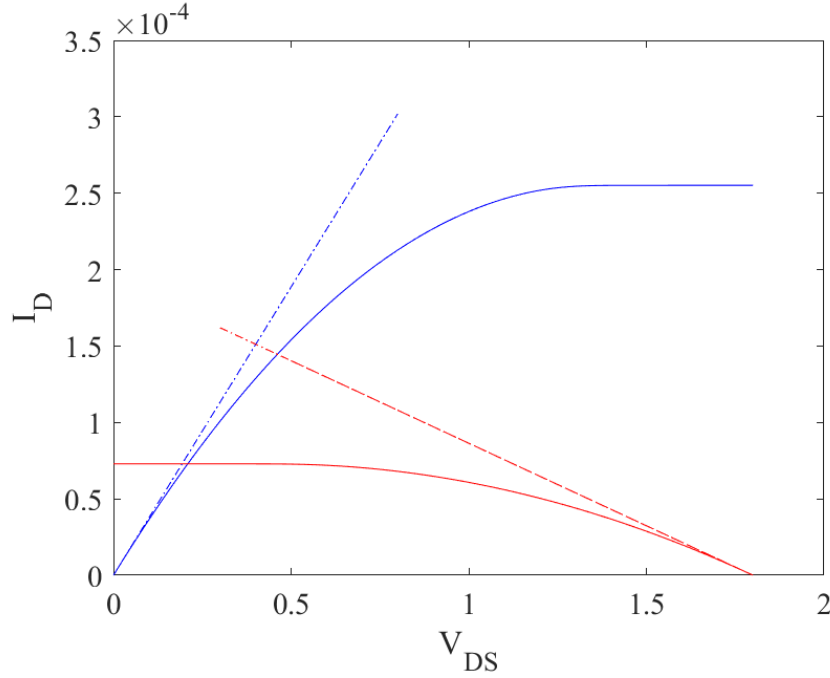


Figure A.4: Balance between a fully conducting pFET and nFET pulling respectively up to Vdd and down to Gnd in a 180nm process with Vdd=1.8V

the separator between weak and strong inversion. In both cases 200nA is below and $2\mu\text{A}$ is above. So all but the top two currents are weak inversion currents.

b)

The drain current when clearly in saturation if $\lambda = 0$ and for given process parameters and given V_{GS} is a constant and may be bigger than that constant if $\lambda > 0$ (for increasing V_{DS}). The strong inversion approximation can be used here to derive that current. Only if the current is smaller than that saturation current, can the transistor be in conduction.

$$I_D = 0.5/1.2 * 3 * 200e - 6 * (0.7 - 0.4)^2 = 22.5\mu\text{A}$$

So the separation is above $10\mu\text{A}$ and all the listed currents can be in conduction.

Solution A.1.11

We are clearly in strong inversion. The holding resistance is the deep conduction r_{ds} , provided that the resulting drain voltage does not deviate too far from $\frac{V_{dd}}{2}$. So let's try. first for 180nm technology:

$$R_{ONn} = r_{dsn} = \frac{1}{k_n V_{OVn}} = \frac{1}{280 \frac{\mu\text{A}}{\text{V}^2} * (1.8\text{V} - 0.45\text{V})} = 2.65\text{k}\Omega \quad (\text{A.17})$$

$$R_{ONp} = r_{dsp} = \frac{1}{k_p V_{OVp}} = \frac{1}{80 \frac{\mu A}{V^2} * (1.8V - 0.45V)} = 9.26k\Omega \quad (A.18)$$

$$V_D = \frac{R_{ONn}}{R_{ONn} + R_{ONp}} = 0.40V \quad (A.19)$$

With that result, the nFET remains in conduction, i.e. $V_D < V_{OVn} = 1.35V$, but the pFET will actually be slightly in saturation as $V_{dd} - V_D > V_{OVp} = 1.35V$. So the real result will be an even lower voltage...

Thus using the saturation current model for the pFET one would get:

$$I_D = \frac{1}{2} k_p V_{OV}^2 = 0.5 * 80 \frac{\mu A}{V^2} * 1.35^2 V^2 = 73\mu A \quad (A.20)$$

Then the voltage across the nFET (who is in deep conduction) is:

$$V_D = I_D R_{ONn} = 73\mu A * 2.65k\Omega = 0.19V \quad (A.21)$$

The use of the full EKV can be done with in a MATLAB script and can illustrate what is wrong with the first result in a plot. The MATLAB script 'EKV.m' computes the full EKV for an nFET:

```
W=1e-6
L=1e-6
UT=26e-3
Vt=0.45
n=1
lambda=0
Vd=0:0.02:1.8;
Vs=0
sty='b'

uCox=80e-6
[Idp]=EKV(W,L,uCox,UT,Vt,n,Vg,Vs,Vd,lambda,sty);
Idp=Idp(size(Idp,2):-1:1)

uCox=280e-6
[Idn]=EKV(W,L,uCox,UT,Vt,n,Vg,Vs,Vd,lambda,sty);
hold on

plot(Vd,Idp,'r')
plot([0,0.8],[0,0.8/2.65e3],'b-.')
plot([0.3,1.8],[1.5/9.26e3,0],'r-.')
print -dpng ../Compendium/fig/BalancePUvsPD180nm.png
```

The resulting plot can be seen in figure A.4. Where the lines cross the current through the nFET and pFET are the same and an equilibrium is reached, so that is where V_D will settle. Zooming in reveals that it is when $V_D = 0.21V$. The discrepancy results from r_{dsn} being underestimated, since we are at a small distance from $V_D = 0$ and the blue curve bends away ever so slightly from a straight line.

The first result is using the deep conduction approximation, so it's where the dash-dotted lines cross. It would only be appropriate if both transistors were clearly in conduction.

The same for 45nm:

$$R_{ONn} = r_{dsn} = \frac{1}{k_n V_{OVn}} = \frac{1}{290 \frac{\mu A}{V^2} * (1.2V - 0.45V)} = 4.60k\Omega \quad (A.22)$$

$$R_{ONp} = r_{dsp} = \frac{1}{k_p V_{OVp}} = \frac{1}{90 \frac{\mu A}{V^2} * (1.2V - 0.45V)} = 14.82k\Omega \quad (A.23)$$

$$V_D = \frac{R_{ONn}}{R_{ONn} + R_{ONp}} = 0.23V \quad (A.24)$$

So here too we find ourselves with a voltage that would imply that the pFET is in saturation even more clearly beyond V_{sat} . So if we again take the saturation approach for pFET we get:

$$I_D = \frac{1}{2} k_p V_{OV}^2 = \frac{1}{2} 90 \frac{\mu A}{V^2} * 0.75^2 V^2 = 25\mu A \quad (A.25)$$

and

$$V_D = I_D R_{ONn} = 25\mu A * 4.60k\Omega = 0.11V \quad (A.26)$$

Still probably underestimating the voltage somewhat because of the deep conduction r_{ds} becomes bigger away from $V_D = 0$ and the real curve is slightly convex. The full EKV graph would look similar.

Solution A.1.12

Using the nominal Vdd one can assume strong inversion and thus equation (4.8) applies. For 180nm and the nFET:

$$I_{ONn} = \frac{1}{4} k V_{dd}^2 \left[\left(\frac{3}{2} - \frac{V_t}{V_{dd}} \right)^2 - \frac{1}{2} \right] \quad (A.27)$$

$$= 0.25 * 280e - 6 * 1.8^2 * ((3/2 - 0.45/1.8)^2 - 1/2) = 241\mu A$$

$$t_{PLH} = \frac{V_{dd} C}{2 I_{ONn}} = 1.8 * 50e - 15 / 2 / 241e - 6 = 187ps$$

$$R_{ONn} = \frac{3 V_{dd}}{4 I_{ON}} = 3/4 * 1.8 / 241e - 6 = 5.60k\Omega$$

$$t_{PLH} = 0.69 R_{ONn} C = 0.69 * 5.60e3 * 50e - 15 = 193ps$$

For a lower Vdd of 0.3V, i.e. $2/3 V_t$, we can use the weak inversion saturation approximation for I_{start} , and in fact also for $I_{\frac{V_{dd}}{2}}$ as with a $V_{DS} = 0.15$ we are still somewhat above $4U_T$, so the saturation current is still a reasonable approximation.

$$I_{\text{start}} = I_S e^{\frac{V_{OV}}{U_T}} = 2 * 290e - 6 * 26e - 3^2 * \exp(-0.15/26e - 3) = 1.22\text{nA}$$

$$I_{\frac{V_{dd}}{2}} = 2 * 290e - 6 * 26e - 3^2 * \exp(-0.30/26e - 3) = 3.82\text{pA}$$

$$I_{ONn} = \frac{I_{\text{start}} + I_{\frac{V_{dd}}{2}}}{2} = 614\text{pA}$$

$$t_{PLH} = \frac{V_{dd}C}{2I_{ONn}} = 0.3 * 50e - 15/2/614e - 12 = 12.2\mu\text{s}$$

$$R_{ONn} = \frac{3V_{dd}}{4I_{ON}} = 3/4 * 0.3/614e - 12 = 366\text{M}\Omega$$

$$t_{PLH} = 0.69R_{ONn}C = 0.69 * 366e6 * 50e - 15 = 12.6\mu\text{s}$$

Solution A.1.13

UT=23e-3

uCox=300e-6

Vt=0.5

lambdaL=0.05e-6

n=1

W=2e-6

L=0.3e-6

k=W/L*uCox

% k = 0.0020

IS=UT^2*k*2*n

% IS = 2.1160e-06

%Id=IS*exp(Vov/(n*UT))

%%

Id=1e-9

Vov=(log(Id)-log(IS))*n*UT

A01=L/(lambdaL*UT)

% A01= 260.87

Id=1e-7

Vov=(log(Id)-log(IS))*n*UT

A02=L/(lambdaL*UT)

% A02= 260.87

%Id=0.5*k*Vov^2

%%

Id=1e-5

Vov=sqrt(2*Id/k)

A03=2*L/(lambdaL*Vov)

% A03 = 120

```

Id=1e-3
Vov=sqrt(2*Id/k)
A04=2*L/(lambdaL*Vov)
% A04 = 12

```

A.2 Linear Circuit Analysis Basics

A.2.1

Draw the Bode plots (both magnitude and phase!) for the following two transfer functions A_1 and A_2 . The x-axis should extend at least 1 decade beyond the highest pole or zero frequency. Use a grid like the one in the illustration A.5! Draw the plots as 'piece-wise linear' approximation of the real graphs, like on the lecture slides: for phase transitions use either $\pm$ one decade (like in FYS3220) or $\pm$ half a decade (like in the graphs on the lecture slides) around the respective pole or zero to complete the full phase shift. For cut-off frequencies in the magnitude plots make an abrupt transition/knee point at the respective pole or zero to change the slope of the magnitude.

$$A_1 = A_{DC1} \frac{1 + \frac{s}{\omega_{z11}}}{\left(1 + \frac{s}{\omega_{p11}}\right) \left(1 + \frac{s}{\omega_{p12}}\right)} \quad (\text{A.28})$$

where $A_{DC1} = 10^3$, $\omega_{z11} = 10^6 \text{rad}$, $\omega_{p11} = 10^3 \text{rad}$, $\omega_{p12} = 10^8 \text{rad}$.

$$A_2 = A_{DC2} \frac{\left(1 + \frac{s}{\omega_{z21}}\right) \left(1 - \frac{s}{\omega_{z22}}\right)}{\left(1 + \frac{s}{\omega_{p21}}\right) \left(1 + \frac{s}{\omega_{p22}}\right) \left(1 + \frac{s}{\omega_{p23}}\right)} \quad (\text{A.29})$$

where $A_{DC1} = 10^4$, $\omega_{p21} = 10^3 \text{rad}$, $\omega_{p22} = 10^7 \text{rad}$, $\omega_{p23} = 10^9 \text{rad}$, $\omega_{z21} = 10^5 \text{rad}$, $\omega_{z22} = 10^7 \text{rad}$

A.2.2

Consider the two Bode plots for transfer functions A_1 and A_2 in Fig. A.6. Find the corresponding transfer functions in root form and Implement the MATLAB function 'function [A1,A2]=transferF(s)' in file transferF.m which computes both transfer functions in dependency of $s = j * \omega$ (but we shall just use the variable s here).

A.2.3

The transfer function of a second order low pass filter has two poles, a known first pole $\omega_{p1} = 100 \text{Rad}$ and an unknown second pole ω_{p2} and no zeros. Its DC gain A_{DC} is 42dB. It shall be used in a negative feedback configuration as introduced in the lecture with a feedback factor $\beta = \frac{1}{10}$. Its phase margin is 60° . Determine the unity gain ω_T of the loop $A\beta$ (!) and the frequency in Rad of the second pole ω_{p2} ! Upload a matlab script in file 'PhaseMargin.m' that assigns the results for respectively ω_T and ω_{p2} to variable 'wt' and 'wp2'!

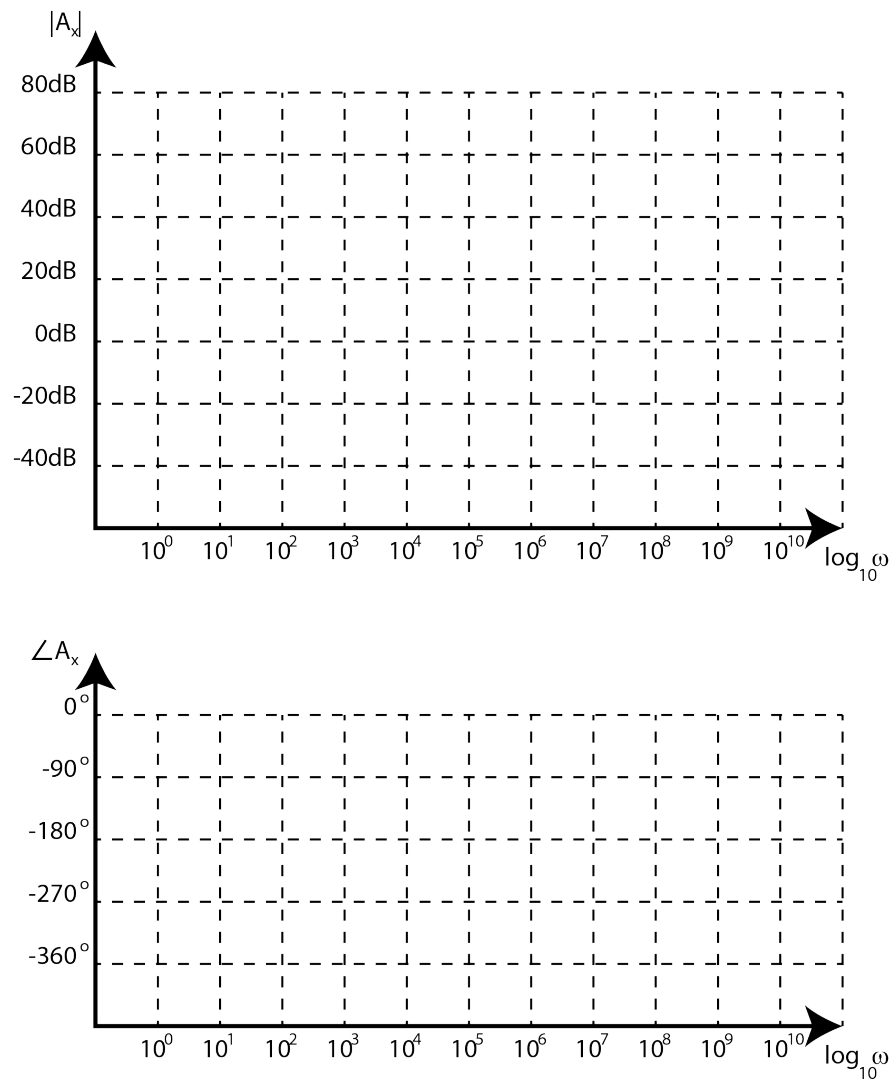


Figure A.5: Template to draw Bode plots in task A.2.1.

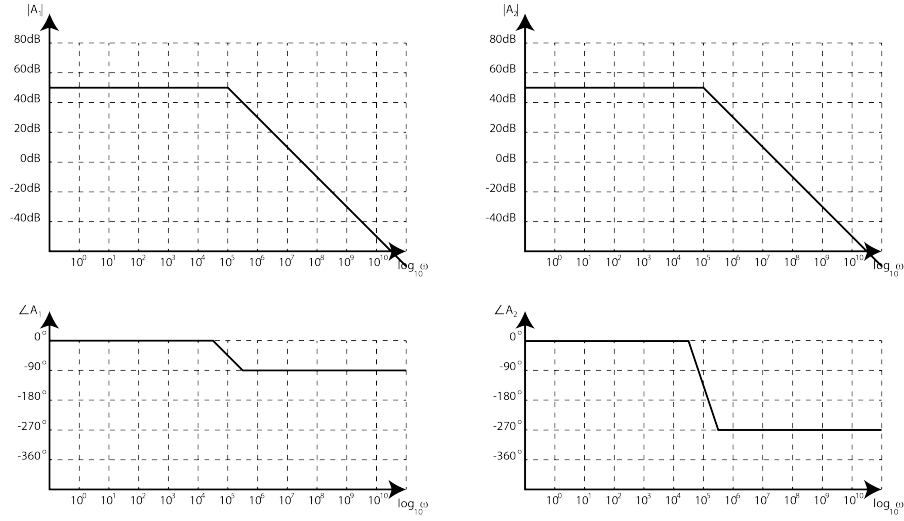


Figure A.6: Bode plots for task A.2.2.

A.2.4

Given the process parameters of table A.1 (this time including n and λ from the table where relevant) of the 180nm process, compute the small signal DC gain of a CS gain stage using an nFET and a load resistance equal to that nFET's drain resistance r_{ds} , biased at $4\mu\text{A}$ for a transistor of width $1\mu\text{m}$ and two different lengths: a) $1\mu\text{m}$ and b) $5\mu\text{m}$.

Then compute the transfer functions for the two cases given that in case a) $C_{GS} = 11\text{fF}$, $L_{OV} = 0.1\mu\text{m}$, and a signal source with an output resistance of $100\text{k}\Omega$, assuming that there is a dominant pole at the input node.

Is the dominant pole at the input node a justifiable assumption if there is a total load capacitance C_L^* of the same magnitude as C_{GS} ?

A.2.5

What are the transfer functions corresponding to the two Bode plots in figures A.7 and A.8? Implement them in a MATLAB function as functions of $s = j\omega$ with header

[H1, H2]=TransferFunctions(s) in file TransferFunctions.m and upload the file.

You may safely assume that any poles or zeros occur only exactly at frequencies that are multiples of 10, and that the DC gain is a multiple of 10 as well.

A.2.6

For a simplified small signal model of a current mirror, determine the transfer function from input current (!) to output current (!), i.e. $\frac{i_o}{i_i}$. For this simplified

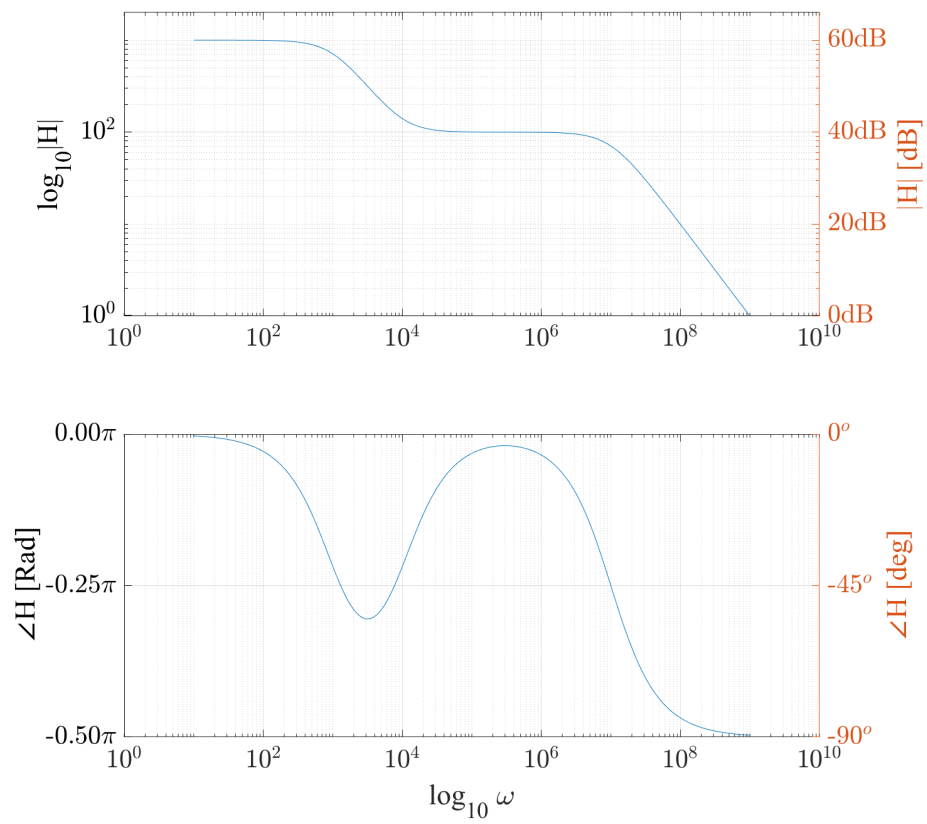


Figure A.7: Bodeplot for transfer function H1 for task A.2.5

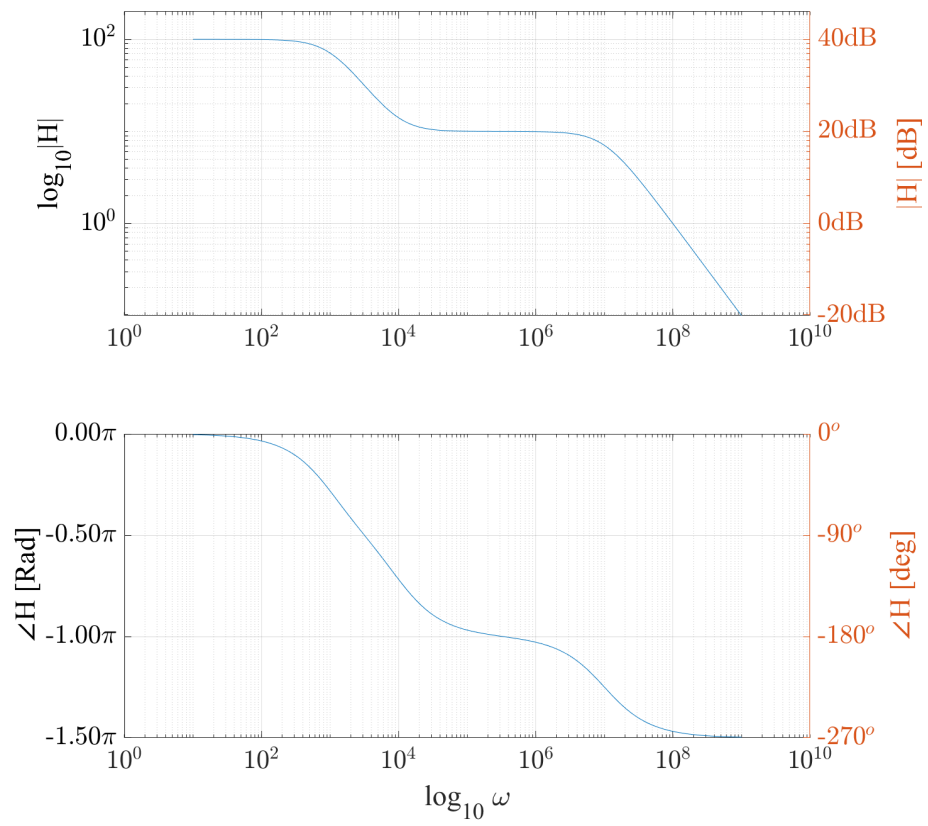


Figure A.8: Bodeplot for transfer function H2 for task A.2.5

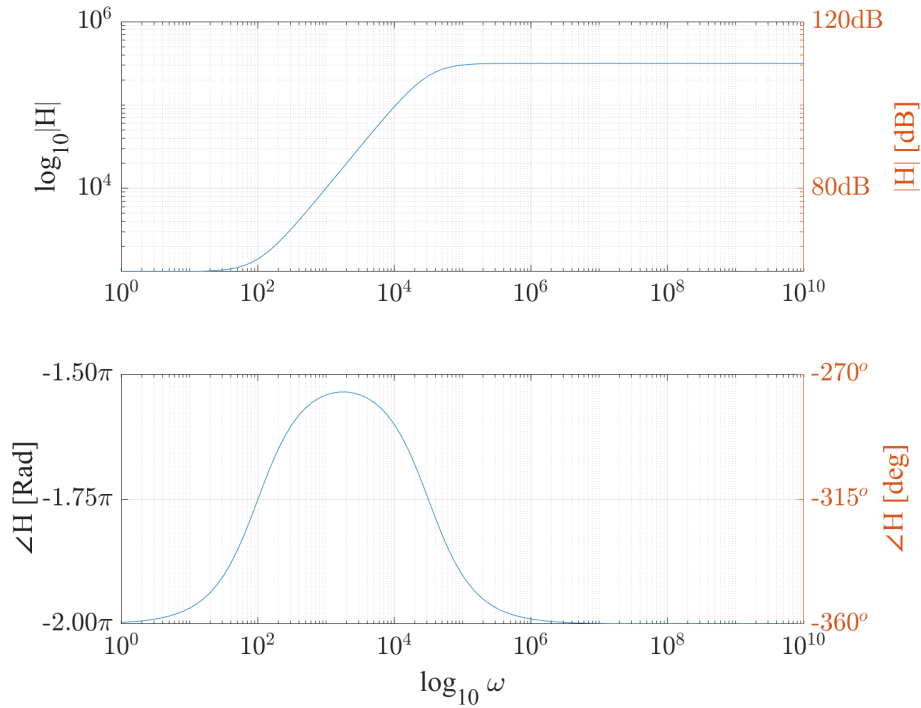


Figure A.9: Bodeplot for transfer function for task A.2.7

model let's ignore any load capacitance or other capacitance of the output node towards Gnd, but include a FB capacitance C_{FB} (which would be C_{GD} of the output nFET), and include a capacitance of the input node towards Gnd (which would be C_{gs} of both transistors and C_{db} of the input transistor). Ignore any r_{ds} of , but include an explicit load resistor R_L between output and small signal Gnd that will conduct the output current.

Then determine an expression for the Q-factor. Advanced question: Looking at this expression, can you reason out whether there is any chance of the Q-factor becoming larger than 0.5 for any combination of C_I , C_{FB} , and R_L ?

A.2.7

The Bode plot in the figure A.9 specifies a transfer function. Can you make a function $H(s)$ in matlab of variable $s = j\omega$ that produces exactly that Bode plot?

Hint: a phase shift of -360° (or -2π in Rad) corresponds to 0° (or just 0 in Rad).

Submit your result as a MATLAB function in file "TransferFunction.m" with function header:

```
function [H]=TransferFunction(s)
```

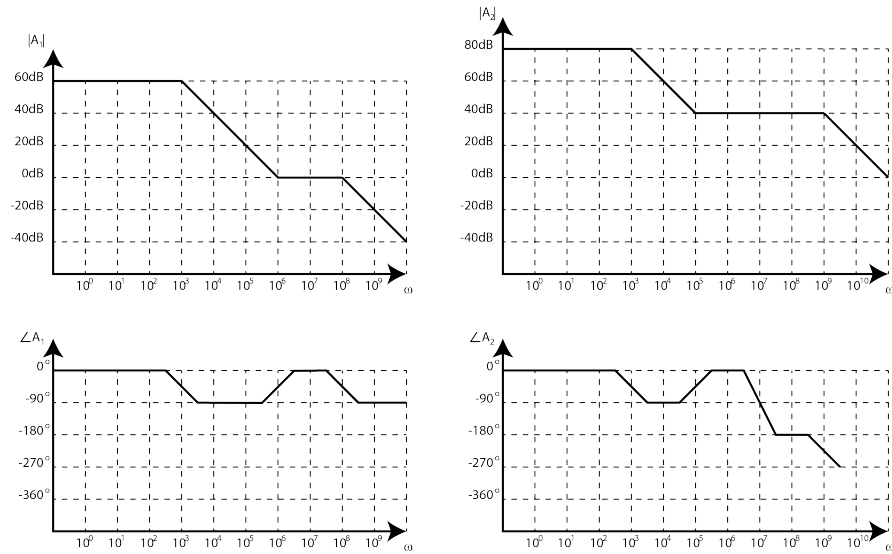


Figure A.10: Solution A.2.1

Solutions section A.2

Solution A.2.1

See figure A.10.

Solution A.2.2

Note that in the following I use the operands with a `'.'` in front, such as `'.^'` and `'.*'`. In matlab this indicates that the operation should be done element-wise on vectors. This allows variable `s` to be a vector of values and the returned values `A1` and `A2` will also be vectors. Thus, one can immediately use for instance `loglogplot(s,abs(A1))'` and get the magnitude plot of `A1`. However, this was not required in the task and you can also just ignore the `'.'`s.

```
function [A1,A2]=transferF(s)

K=10.^(50./20)
A1=K.*1./(1+s./10.^5);
A2=A1.*(1-s./10.^5)./(1+s./10.^5);

end
```

Solution A.2.3

```
wp1=100;
beta=0.1;
wt=wp1*10^(42/20)*beta
```

% for 60deg PM from table in lecture:
% wt/wp2=0.580

wp2=wt/0.58

Solution A.2.4

a)

$$I_S = 2nkU_T^2 = 2 * 1.6 * 1 * 280e - 6 * 26e - 3^2 = 0.606\mu A \quad (A.30)$$

So it's ok to use strong inversion equations.

b) (5 times smaller I_S)

$$I_S = 2nkU_T^2 = 2 * 1.6 * 0.2 * 280e - 6 * 26e - 3^2 = 0.121\mu A \quad (A.31)$$

And even more clearly strong inversion

Defining $V_O = \frac{V_{dd}}{2}$ we can also assume saturation.

So:

a)

$$g_m = \sqrt{2kI_D} = \text{sqrt}(2 * 280e - 6 * 4e - 6) = 47.3 \frac{\mu A}{V} \quad (A.32)$$

$$r_{ds} = \frac{1}{\lambda I_D} = \frac{L}{L\lambda I_D} = 1e - 6 / 0.08e - 6 / 4e - 6 = 3.125M\Omega \quad (A.33)$$

Gain with a load of equal resistance to r_{ds} is

$$A_{DC} = -g_m \frac{r_{ds}}{2} = 47.3e - 6 * 3125e3 / 2 = -73.9 \quad (A.34)$$

b) g_m will be smaller by 1 over sqrt of 5 and r_{ds} 5 times bigger so the gain sqrt 5 times bigger.

$$g_m = \sqrt{2kI_D} = \text{sqrt}(2 * 0.2 * 280e - 6 * 4e - 6) = 21.2 \frac{\mu A}{V} \quad (A.35)$$

$$r_{ds} = \frac{1}{\lambda I_D} = \frac{L}{L\lambda I_D} = 5e - 6 / 0.08e - 6 / 4e - 6 = 15.625M\Omega \quad (A.36)$$

$$A_{DC} = -g_m \frac{r_{ds}}{2} = 21.2e - 6 * 15625e3 / 2 = -165.6 \quad (A.37)$$

a)

$$C_{GD} = WL_{OV}C_{OX} = L_{OV} \frac{C_{GS}}{(L_{OV} + \frac{2}{3}L)} = 0.1\mu m \frac{11fF}{(0.1 + \frac{2}{3})\mu m} = 1.43fF \quad (A.38)$$

$$\frac{v_o}{v_{sig}} = A_{DC} \frac{1}{1 + sR_{SIG}(C_{GS} + C_{GD}(1 - A_{DC}))} = 73.9 \frac{1}{1 + s * 100e3 * 118e-15} \quad (\text{A.39})$$

(Note the negative A_{DC} , so that's why we write $(1 - A_{DC})$ for the Miller cap.)

So a time constant of 11.8ns and a cut off frequency in Rad of 84.74MRad

Actually the approximation with the input node being the place of the dominant pole i.e. only the input node being frequency dependent does not hold: with an output resistance of more than 10 times larger than the signal source, i.e. 1.5625M Ω , and a capacitance more than a 10th of the input cap with Miller effect of 11fF, the assumption of the dominant pole at the output would yield a lower cut off frequency of 58MRad. So assuming a dominant pole at the input in this case is actually wrong!

b)

C_{GD} is the same but $C_{GS} = 51\text{fF}$.

$$\frac{v_o}{v_{sig}} = A_{DC} \frac{1}{1 + sR_{SIG}(C_{GS} + C_{GD}(1 - A_{DC}))} = 165.6 \frac{1}{1 + s * 100e3 * 289.2e-15} \quad (\text{A.40})$$

So a time constant of 28.9ns and a cut off frequency in Rad of 34.6MRad

Assuming a dominant pole at the output would yield a cut off at $1/(15.625e6 * 51e-15) = 1.25\text{MRad}$, so also here and even more clearly the assumption of a dominant pole at the input is actually wrong.

Solution A.2.5

The two Bode plots are resulting from the following transfer functions:

$$\begin{aligned} H1 &= 1000 \frac{(1 + \frac{j\omega}{10^4})}{(1 + \frac{j\omega}{10^3})(1 + \frac{j\omega}{10^7})} \\ H2 &= 100 \frac{(1 - \frac{j\omega}{10^4})}{(1 + \frac{j\omega}{10^3})(1 + \frac{j\omega}{10^7})} \end{aligned}$$

As matlab function:

```
function [H1,H2] = TransferFunctions(s)

H1= 1000*(1+s/10^4)./((1+s/10^3))./((1+s/10^7))
H2= 100*(1-s/10^4)./((1+s/10^3))./((1+s/10^7))

end
```

Solution A.2.6

The Kirchoff equations:

$$v_i (g_m + sC_I + sC_{FB}) = i_i + v_o sC_{FB} \quad (\text{A.41})$$

$$v_o \left(\frac{1}{R_L} + sC_{FB} \right) + v_i g_m = v_i sC_{FB} \quad (\text{A.42})$$

$$v_o = i_o R_L \quad (\text{A.43})$$

From there:

$$v_i = v_o \frac{\frac{1}{R_L} + sC_{FB}}{sC_{FB} - g_m} = i_o R_L \frac{\frac{1}{R_L} + sC_{FB}}{sC_{FB} - g_m} \quad (\text{A.44})$$

$$i_o R_L \frac{\left(\frac{1}{R_L} + sC_{FB} \right) (g_m + sC_I + sC_{FB}) - sC_{FB} (sC_{FB} - g_m)}{sC_{FB} - g_m} = i_i \quad (\text{A.45})$$

$$\begin{aligned} \frac{i_o}{i_i} &= \frac{1}{R_L} \frac{sC_{FB} - g_m}{\left(\frac{1}{R_L} + sC_{FB} \right) (g_m + sC_I + sC_{FB}) - sC_{FB} (sC_{FB} - g_m)} \\ &= \frac{s \frac{C_{FB}}{g_m} - 1}{1 + s \left(\frac{C_I + C_{FB}}{g_m} + 2R_L C_{FB} \right) + s^2 \frac{R_L}{g_m} C_I C_{FB}} \end{aligned} \quad (\text{A.46})$$

To get Q

$$\omega_0^2 = \frac{1}{\frac{R_L}{g_m} C_I C_{FB}} \quad (\text{A.47})$$

$$Q = \frac{\sqrt{\frac{R_L}{g_m} C_I C_{FB}}}{\frac{C_I + C_{FB}}{g_m} + 2R_L C_{FB}} \quad (\text{A.48})$$

can this become bigger than 0.5? In the compendium we us a couple of times that

$$\frac{\sqrt{ab}}{a+b} < \frac{1}{2} \quad (\text{A.49})$$

because 1)

$$a = b \Rightarrow \frac{\sqrt{ab}}{a+b} = \frac{\sqrt{a^2}}{2a} = \frac{1}{2} \quad (\text{A.50})$$

and 2)

$$b = a + \epsilon \Rightarrow \frac{ab}{(a+b)^2} = \frac{a(a+\epsilon)}{(2a+\epsilon)^2} = \frac{a^2 + a\epsilon}{4a^2 + 4a\epsilon + \epsilon^2} < \frac{1}{4} \quad (\text{A.51})$$

So if you take $a = R_L C_{FB}$ and $b = \frac{1}{g_m} C_I$ you can say that

$$Q = \frac{\sqrt{\frac{R_L}{g_m} C_I C_{FB}}}{\frac{C_I + C_{FB}}{g_m} + 2R_L C_{FB}} < \frac{\sqrt{\frac{R_L}{g_m} C_I C_{FB}}}{\frac{C_I}{g_m} + R_L C_{FB}} < \frac{1}{2} \quad (\text{A.52})$$

For those very interested ;-)) it could be an exercise to include a capacitor C_L at the output and to figure out, whether it is still the case that there is never any resonance...

Solution A.2.7

```
function [H]=TransferFunction(s)

p1=10^4.5;
z1=10^2;
Adc=10^3;

H=Adc*(1+s./z1)./(1+s./p1)
```

A.3 Digital Circuit Basics**A.3.1**

simplify:

$$a \vee a \wedge b \quad (\text{A.53})$$

$$a \wedge a \vee b \quad (\text{A.54})$$

$$(a \vee a) \wedge b \quad (\text{A.55})$$

$$a \wedge (a \vee b) \quad (\text{A.56})$$

A.3.2

Find equivalent expressions that only use the inverted variables as inputs (useful to derive a PU network of pFETs from a PD network of nFETs).

Example:

$$\overline{a \vee b} = \bar{a} \wedge \bar{b} \quad (\text{A.57})$$

$$\overline{a \vee b \vee c} = ? \quad (\text{A.58})$$

$$\overline{a \vee b \wedge c} = ? \quad (\text{A.59})$$

$$\overline{(a \vee b) \wedge c} = ? \quad (\text{A.60})$$

Now derive an equivalent expression that is the inverse of an expression with only non-inverted variables as inputs (useful to derive a PD network of nFETs from a PU network of pFETs).

Example:

$$\bar{a} \vee \bar{b} = \overline{a \wedge b} \quad (\text{A.61})$$

$$\bar{a} \vee \bar{b} \vee \bar{c} = ? \quad (\text{A.62})$$

$$\bar{a} \vee \bar{b} \wedge \bar{c} = ? \quad (\text{A.63})$$

$$(\bar{a} \vee \bar{b}) \wedge \bar{c} = ? \quad (\text{A.64})$$

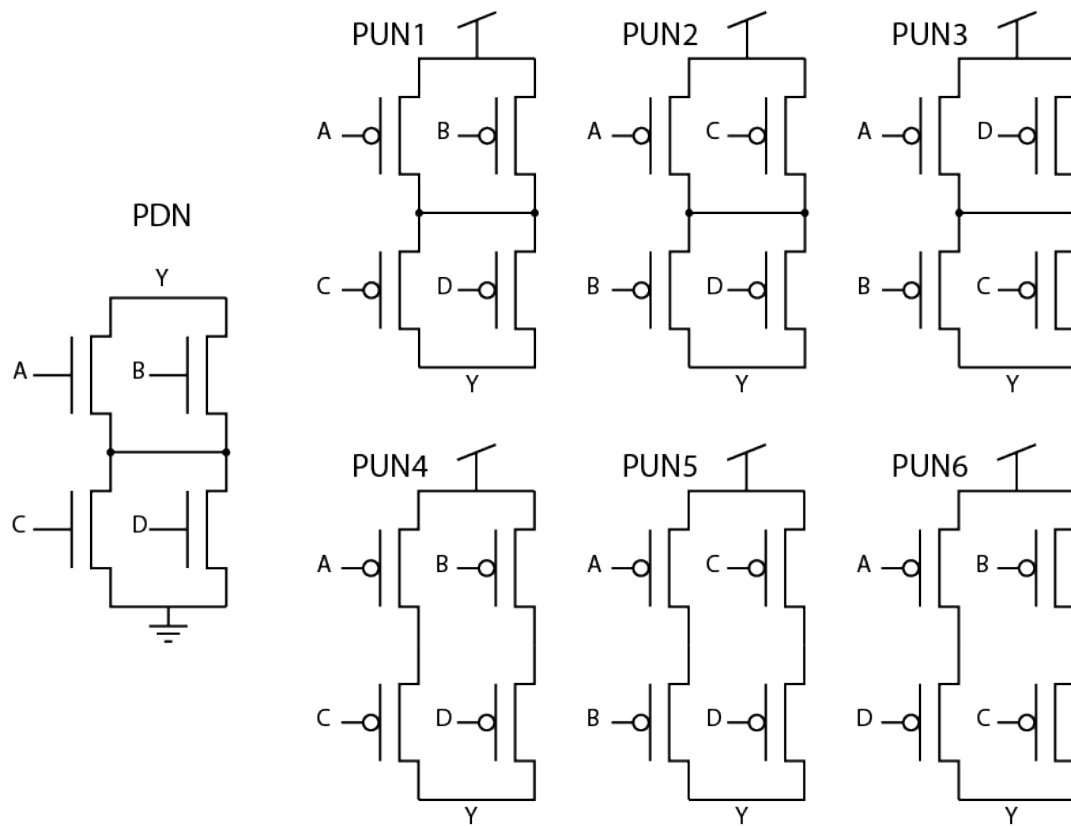


Figure A.11: PDN and corresponding PUN candidates. Illustration task A.3.4

A.3.3

Implement a static two input AND gate (Note: not a NAND gate!) with complementary PUN and PDN and input signals A and B , using nFETs and pFETs, power supply Vdd and Gnd.

A.3.4

Which of the pull up networks (PUN) in figure A.11 would complement the pull down network (PDN) shown. Multiple answers may be possible.

A.3.5

Please order the digital gates in figure A.12 by their propagation delays (fastest on top, slowest on the bottom) in a particular situation: specifically the delay for output signal Y to switch after the input signal A switches from low to high while input signals B and C are, and remain, high and signal D is ,and remains, low. The transistors have all the same $\frac{W}{L}$ ratio and C_L is the same for all gates.

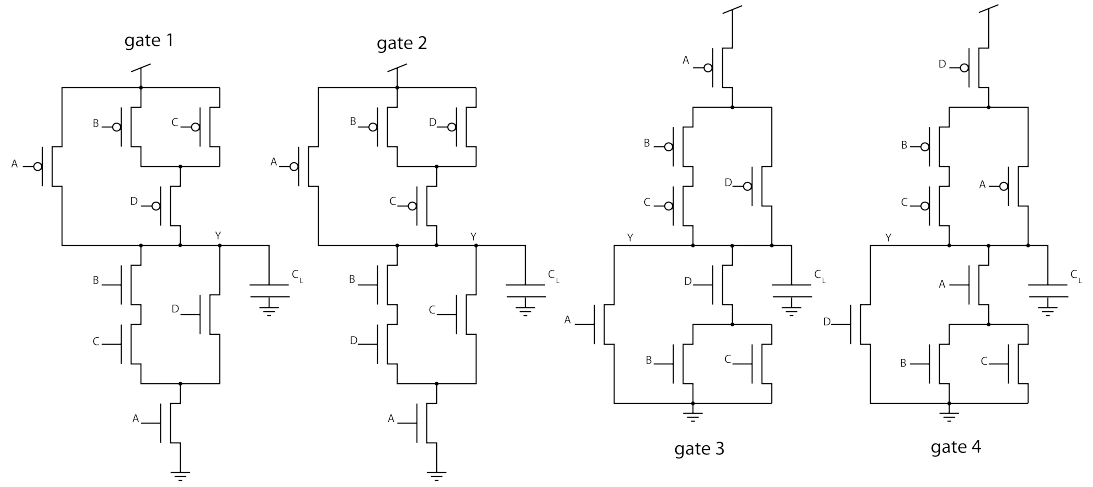


Figure A.12: Digital gates for task A.3.5

Should two gates have the same delay, just order them next to each other and it does not matter which one you place before the other.

A.3.6

Which of the gates in figure A.13 switches the fastest when all inputs (A;B;C) go from low to high, and which is second, third and slowest? Consider the 'internal' capacitors on the output node (!) only as listed in the compendium according to equation (7.3), as well as the equivalent resistances of the conducting transistors. Note that the letters 'p' and 'n' indicate the W/L ratio of a balanced inverter, i.e. where both the PUN and PDN of that inverter have the same equivalent resistance when the transistor is conducting/switched on. So here, some of the transistors have double the width W at the same length and thus double the W/L ratio of that inverter. Here we do not consider any load capacitor, neither from the wire or from any subsequent circuit, so only the parasitic capacitors of the transistors connected to the output node Y and all equivalent resistances of conducting transistors.

A.3.7

If an inverter in a certain CMOS technology has balanced propagation delays $t_{PHL} = t_{PLH}$ and the width to length ratio for the pull down nFET is $\frac{W_n}{L_n} = n$ and for the pull up pFET is $\frac{W_p}{L_p} = p$, which of the logic gates in figure A.14 are appropriately balanced, i.e. do have the same worst case approximate current drive (i.e. neither less nor more) in both the HL and LH transitions as that inverter? The transistors do have the same lengths as the inverter's but their width is changed. The ratio x of their widths relative to the inverter's is indicated as xp and xn in the figure.

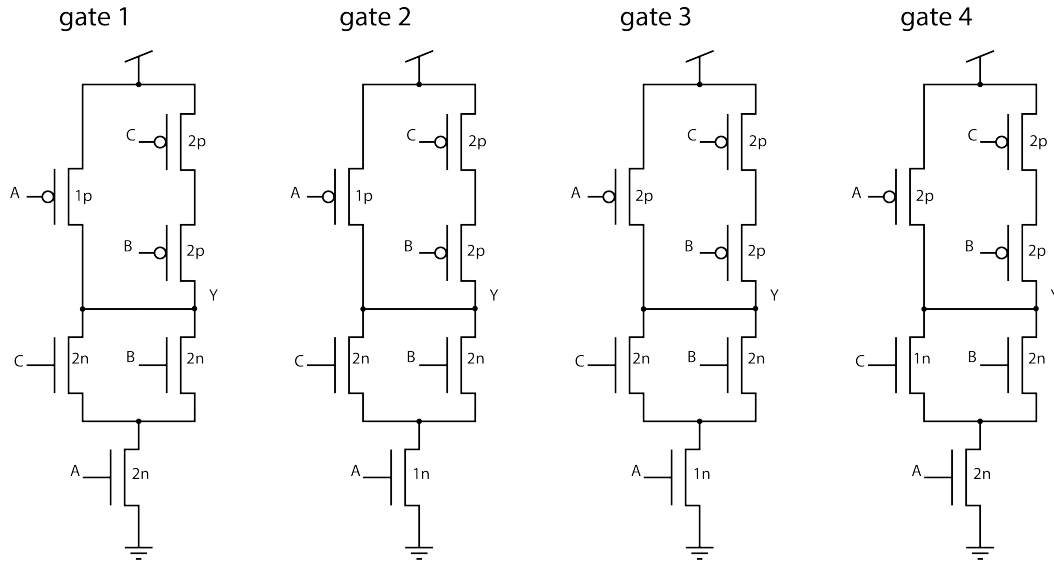


Figure A.13: Digital gates for task A.3.6

A.3.8

The gates in the illustration A.15 are designed to have all the same balanced equivalent on-resistance R_{ON} as an inverter. Balanced means that R_{ON} is the same both for pulling the output up and down. All transistors used in the gates have the same length L , but they may have different widths W to make R_{ON} the same as for an inverter and to make R_{ON} balanced. The mobility of nFETs in this process is 2.5 times higher than for pFETs, so for an inverter the pFET's width W is 2.5 times wider than the nFET's. Let us assume that the total capacitance in the nodes is only composed of the input capacitance C_I of the logic gates and that that capacitance is entirely proportional to the FET gate area it is connected to. Please order the nodes a) to f) marked in the illustration according to their worst case signal delay, i.e. the worst accumulated propagation delay for the signal path from any of the inputs. Assign exactly one position in the ranking to all inputs. Should there be an equal delay between multiple nodes, their order relative to each other does not matter.

A.3.9

Draw an NP Domino logic schematic implementing the expression

$$Y = (A \vee B) \wedge C \quad (\text{A.65})$$

Use nFETs and pFETs, a Vdd and Gnd power supply and static (!) input signals A, B, C and their inverse $\overline{A}, \overline{B}, \overline{C}$, as well as a clock signal φ and its inverse $\overline{\varphi}$. The precharge phase happens when φ is low and the evaluation phase when φ is high. Use footed logic gates! Mark the output node Y ! The function Y with three input variables shall not be implemented with a single

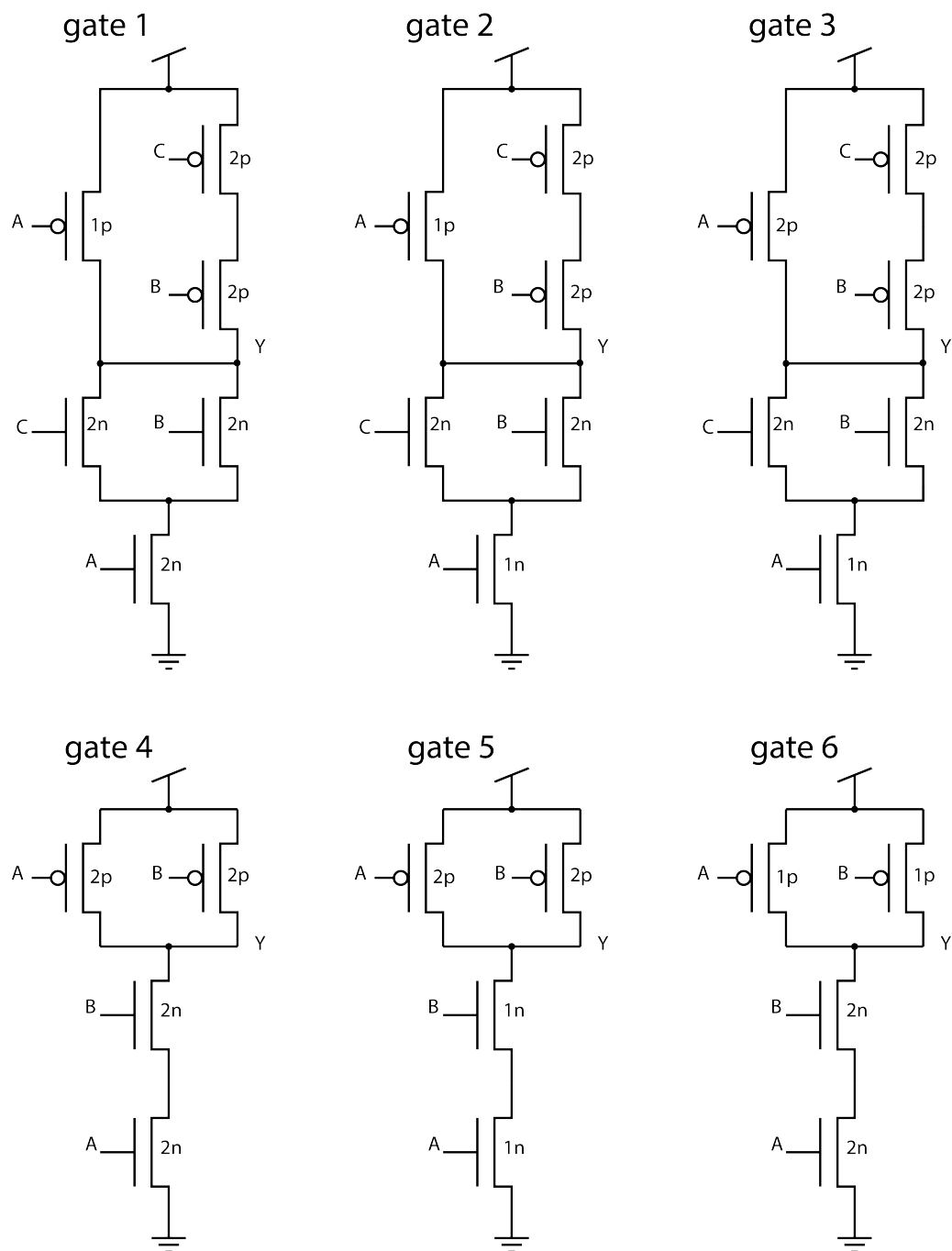


Figure A.14: Digital gates for task A.3.7: which do have balanced PU/PD drive?

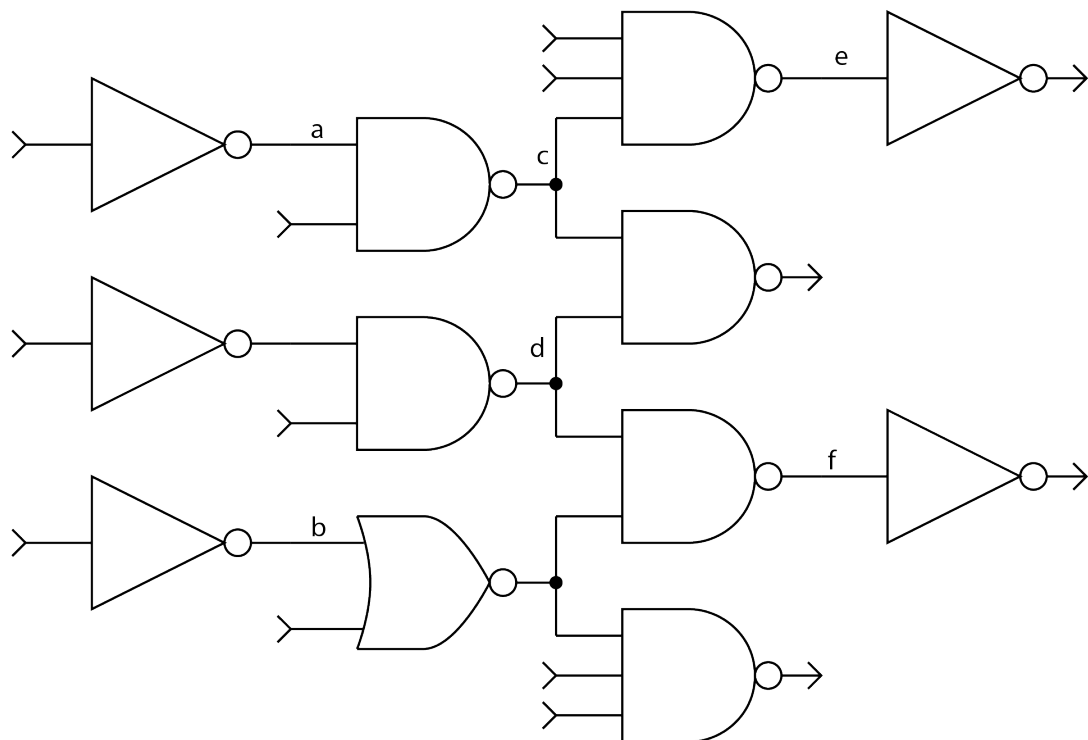


Figure A.15: Combinational logic circuit for task for task A.3.8.

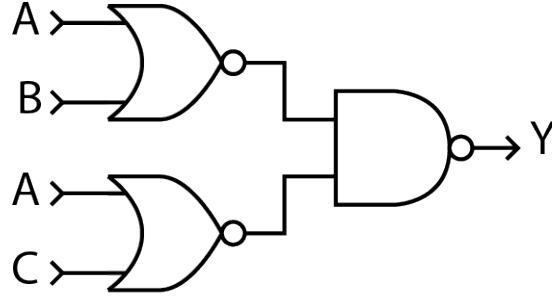


Figure A.16: Logic to be implemented as NP domino in task A.3.10.

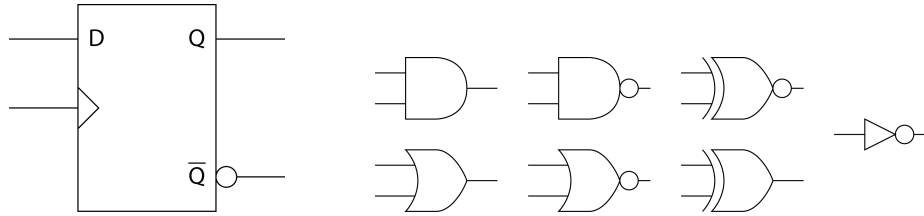


Figure A.17: Symbols to use in task A.3.11. From left to right: DFF, AND, OR, NAND, NOR, XNOR, NOR, NOT.

logic gate with three inputs, but connect multiple gates with just two inputs each together. You might want to use the de'Morgan theorem to transform some of the expression into a form that lends itself more easily to be implemented in NP Domino logic.

A.3.10

Re-implement the gate level schematics in figure A.16 as NP domino logic schematic on transistor level (!), i.e. using pMOS and nMOS transistors as building blocks. Do not attempt to simplify the logic, but implement it 1:1 as NP domino logic with the same three gates. Let's define a clock signal φ and its inverse $\bar{\varphi}$ and that $\varphi = 0 \Leftrightarrow$ precharge and $\varphi = 1 \Leftrightarrow$ evaluation. You may assume that the input signals also come from appropriate NP domino logic (i.e. no footer necessary). Make sure to use the same signal label as in the figure. Make a clear drawing!

A.3.11

Make a JK-flipflop using one D-flipflop and combinational logic gates. A JK-flipflop is a synchronous flip flop with two inputs J and K (as well as the clock signal CLK) and two outputs Q and its inverse $\bar{Q}$. If only J is high, Q will go high and $\bar{Q}$ low at the next rising clock flank. If only K is high, Q will go low

and $\overline{Q}$ will go high at the next rising clock flank. If neither J nor K are high, the previous state is maintained. If both J and K are high the next state will be the inverse of the previous state, i.e. the state will 'toggle'. It is not important that you use a minimal number of gates.

Use symbols for DFF and gates from figure A.17.

A.3.12

Design a simple synchronous (i.e. there is a CLK signal that you do not need to represent explicitly, and state transition only occur synchronously with the rising edge of that clock signal) Moore-type finite state machine (FSM) to control a 'smart' burglar alarm. It shall have three states: 'asleep', 'attentive', and 'alarm'. It receives two binary sensor inputs: 'Movement' and 'human intruder'. Its default state is 'asleep' where it consumes very little energy and only motion sensors are active. They activate signal 'M' which is 0/false when there is no movement and 1/true if there is movement. If movement is detected the system enters its 'attentive' state, turning on a power consuming intelligent sensor system that processes camera images to detect the shape of a human being and distinguishes it from motion triggered by animals or by plants moved by the wind. Humans are signalled by the signal 'H' being 1/true and the system goes to state 'alarm'. On the other hand, if no human shape is detected, 'H' is 0/false and the system goes back to state 'asleep' until the next time it detects motion. An alarm remains active as long as there continues to be motion. If the movement stops, the system goes back to 'asleep'. 1) Draw a state transition graph for this FSM. In the state 'bubbles', do indicate the name and the binary encoding (!) of that state. Attention: if there are fewer states than your binary encoding would allow, make sure to include those 'undefined' states as well with an unconditional transition to the 'asleep' state: that's always good policy in case your circuit ends up in this undefined states by mistake, e.g. during power up of your circuit! 2) Make a state transition table with columns for the 'state now' with as many bits as necessary, with the two one bit input variables H and M, and with the 'next state'. Use binary variable values, 0, 1, and you may use an X for "don't care" states/inputs to simplify. Make sure to include all possible states and inputs in the table.

A.3.13

Design the state transition graph for a synchronous Moore finite state machine implementing an exponential decay counter: The output 'O' of that counter shall be a binary 4 for the first clock cycle, turn to 2 the next clock cycle, then turn to 1 after a total two (!) clock cycles, and finally to 0 after another four (!) clock cycles. It shall hold 0 for another 8 clock cycles, before restarting the whole cycle at output 4 again. It shall have a reset input 'RES' which can set it back to the first state with output 'O=4' at the onset of the next clock cycle. Number your states with a state variable 'S'.

Make a clear drawing!

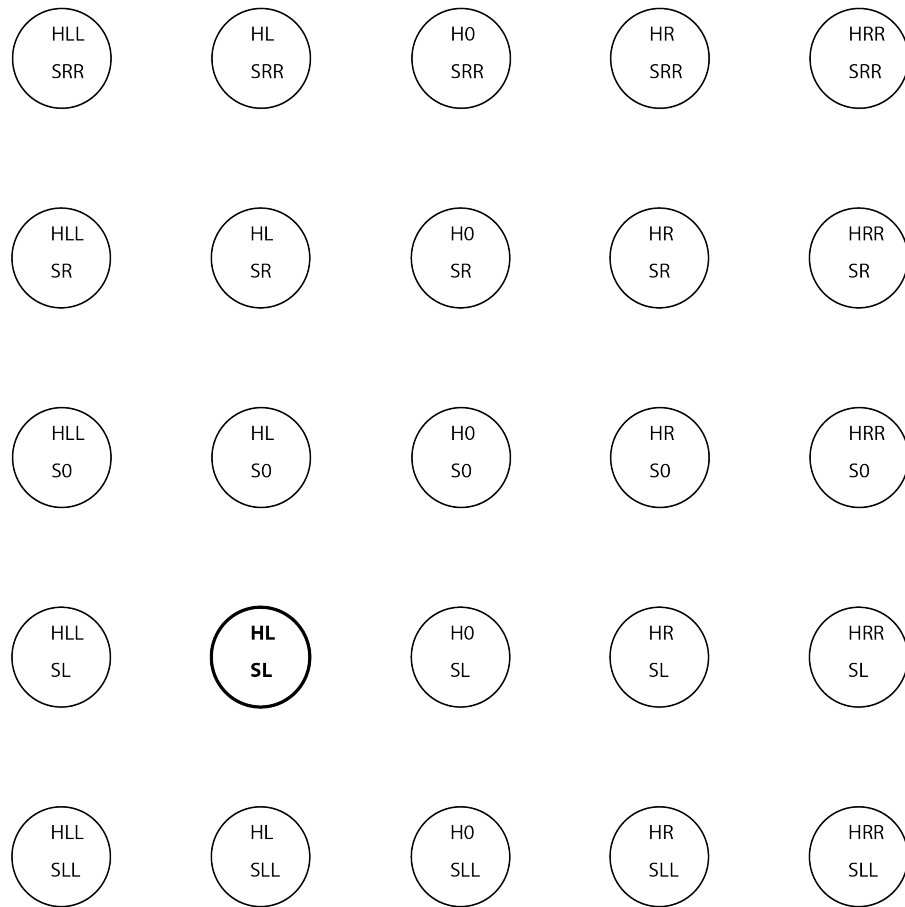


Figure A.18: Representations of the 25 states of the FSM described in task A.3.14

A.3.14

Design a small part of a finite state machine (FSM) for an auto pilot for a ship that has to hold a heading, for example due north. In the following you will only have to supply the state transitions in the state transition graph for the one highlighted state bubble in the collection of state bubbles in the illustration.

Sensor input:

The system has a sensor that indicates deviations from the intended course/heading as binary signals that we'll call H0, HL, HLL, HR, and HRR (the 'H' stands for 'heading'). **Exactly one of these binary sensor signals will be 1 at a time!** All others will be zero.

If the course is correct with less than $\pm 5^\circ$ deviation, then H0=1.

HL=1 indicates that the course deviates by more than 5° but less than 10° to the left and HR=1 indicates a deviation of more than 5° but less than 10° to the right.

HLL=1 indicates that the course deviates more than 10° to the left and HRR=1 indicates a deviation of more than 10° to the right.

Control output:

Likewise we have five control signals that turn the ship's rudder that we'll call S0, SL, SLL, SR, and SRR (the 'S' stands for 'steer'). **Exactly one of them at a time is equal to 1!** So it's OK to write in your state 'bubbles' SL=1 or just SL (see illustration!) as the output signal produced by that state, for example, and all other outputs are implicitly set to zero.

S0=1 means that the rudder is set to steer straight ahead.

SL=1 indicates that the rudder is turned to correct the course towards the left, halfway to the most extreme setting, and SR=1 correspondingly to the right.

SLL=1 indicates that the rudder is turned to correct the course to the left in its most extreme setting, i.e. the ship turns more rapidly than when SL=1, and SRR=1 correspondingly to the right.

Now current and wind and imprecision in the rudder design may make it so that just leaving the rudder in its straight position will not do the job to keep the ship on a straight path. It may be pushed slowly or rapidly of course and course corrections may become necessary.

Autopilot algorithm:

Course corrections (i.e. the next rudder setting) will depend on the rudder setting right now and the sensor input as follows:

a)

If the heading (according to the precision of the sensor) has changed since the last clock cycle, the rudder is turned more towards the opposite direction of that change, and it is turned one 'notch' per 'notch' of that change (limited by the maximal positions of the rudder). So for example if the heading in the previous clock cycle has been HL and the sensor now indicates HR, then the course has changed towards the right by two 'notches'. So if the rudder has been at S0 it should now be turned to SLL. **These changes of the rudder position are accumulated together with what rule b) indicates.**

b)

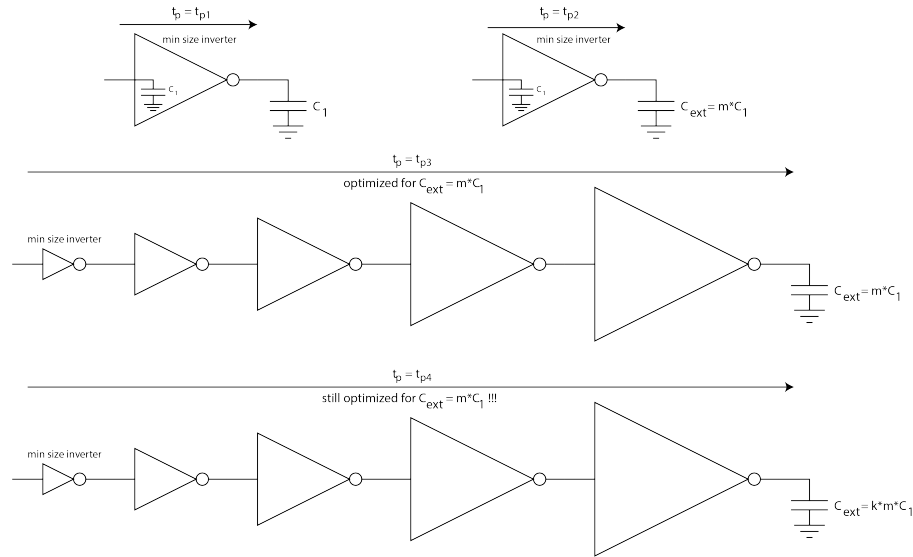


Figure A.19: Multi-inverter digital buffer illustration for task A.3.15

In addition, if the heading HAS BEEN off on the left, no matter whether HL or HLL, add one 'notch' steering to the right (limited by the maximum rudder setting), and if the heading HAS BEEN off to the right, add one 'notch' steering to the left (limited by the maximum). So for example if the heading has been HR and the rudder has been at SR it should be turned another 'notch' to steering left, i.e. to S0. **Note that rule a) could add to that!**

The FSM that controls the rudder needs 25 states. Simply having 5 states for the 5 possible outputs won't do: the states also need to keep track of the last sensor reading to be able to follow the algorithm. The 25 states can be identified by the output to the rudder and the last sensor reading, as depicted in the illustration A.18.

Finally, here is your task:

Draw all possible state transition arrows from only **the one state (SL,HL)** (marked in the illustration A.18) that implement the algorithm as described. You only have to draw that state and its outgoing arrows, as well as all its possible 'descendants'. Mark the arrows with the condition of the input signal for the transition. Hint: ask yourself how many arrows there need to be. Also ask yourself in which subgroup of states each of those arrows can end up in.

A.3.15

Assume a typical minimum size inverter provides a signal to an off-chip capacitance. That off-chip capacitance C_{ext} is m times larger than the input capacitance C_1 of that typical minimum size inverter. If the inverter were to drive another inverter of the same size, it would suffer from a gate delay t_{p1} (' $tp1$ ' in MATLAB). See illustration in figure A.19. Let us use the gate delay model where that inverter drives its output via an on-resistance R_{on1} proportional to

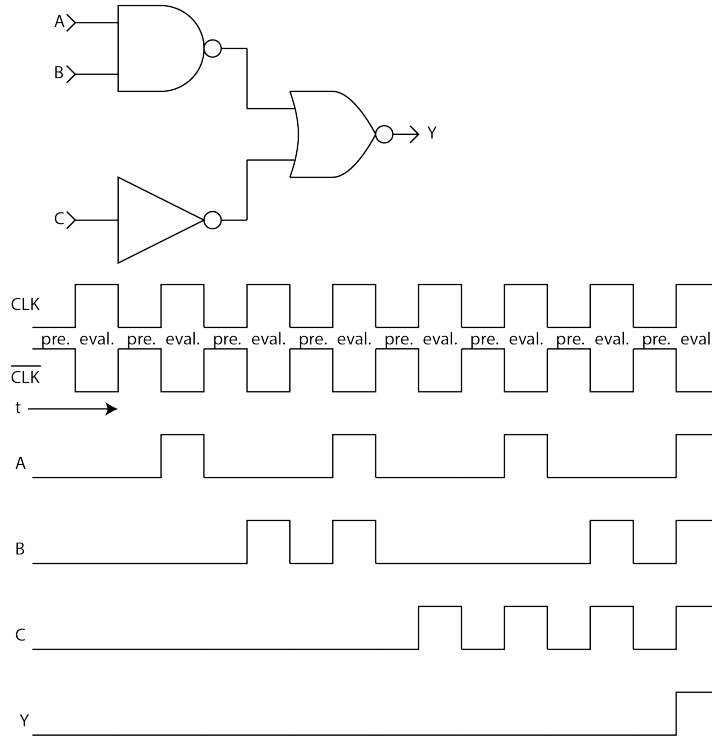


Figure A.20: Gates to be implemented as NP Domino logic in task A.3.16

the inverters' width to length ratio, and let's take only gate input capacitances and the final capacitance C_{ext} into account.

In terms of m and t_{p1} , what is the gate delay t_{p2} (use 'tp2' in MATLAB) if the minimum inverter drives C_{ext} ?

Now, when adding four inverters in series between the initial minimum size inverter and C_{ext} (so including the initial inverter, it will be 5!), to what value t_{p3} (use 'tp3' in MATLAB) can you reduce the delay t_{p2} when sizing the transistor widths of each of those additional four inverters optimally (then length remains the same for all transistors)?

And finally, it is not always possible to predict off-chip loads correctly. So let's assume that the prediction of C_{ext} is off by a factor k , i.e. C_{ext} is not m times bigger than C_1 , but instead $k \cdot m$ times bigger. In that case, keeping the same sizing of the 4 inverters' transistors as optimized for m , what is the total gate delay t_{p4} (use 'tp4' in MATLAB)?

Submit the solution as a Matlab function in a file PropagationDelay.m with the function header:

```
function [tp2,tp3,tp4]=PropagationDelay(tp1,m,k)
```

A.3.16

Implement the combinational logic circuit in the illustration A.20 as NP Domino dynamic logic as a transistor level schematics, i.e. using nFETs and pFETs. Do

NOT simplify but implement it using the three gates as indicated. The input traces come from appropriate NP Domino logic circuits too, as indicated in the signal traces below the schematic. That means they are in a default low (!) state during the precharge phase (marked as 'pre.' on the clock traces) Check the traces in the illustration! This default precharge state determines whether the first gates are N or P type. Then they assume their logic signal level during the evaluation phase (marked as 'eval.' in the figure). The traces undergo all possible combinations of the input signals. The correct output signal trace is also shown. You may use all the signals indicated in your schematics, i.e. the inputs as well as CLK and the inverted CLK, plus Vdd and Gnd as power rails.

A.3.17

Draw a synchronous state transition graph for a (Moore) finite state machine (FSM) that controls an automated car wash.

The user selects between two programs: 'Standard' and 'Delux' which are encoded as a binary input signal P, where P=0 corresponds to 'Standard' and P=1 to 'Delux'. This selection remains locked and unchanged after a washing sequence is started. There is also a start signal 'S' that is high when sufficient money has been inserted and a car is in position. Money will be deducted when a washing sequence starts.

The outputs of your FSM are active high binary variables 'W' for 'wash', 'D' for 'dry', and 'T' for a 'wax finish'.

The sequenc for a 'Standard' wash is simply 'wash' and then 'dry', each for one 'clock cycle'.

The sequence for the 'Delux' wash is a double 'wash', i.e. holding 'W' for two cycles, then a 'wax finish' for one cycle, and finally a 'dry' cycle.

Hint: remember that the output signals need not be the same as the internal state registers but can be derived from them by combinational logic, i.e. you can have the same outputs in different states.

Mark each bobble with a unique state number (integer or binary) and the appropriate output signal states, and the state transition arrows with the appropriate conditions. Arrows with no conditions are used for unconditional transitions.

Solutions section A.3

Solution A.3.1

$$a \vee a \wedge b = a \quad (\text{A.66})$$

$$a \wedge a \vee b = a \vee b \quad (\text{A.67})$$

$$(a \vee a) \wedge b = a \wedge b \quad (\text{A.68})$$

$$a \wedge (a \vee b) = a \quad (\text{A.69})$$

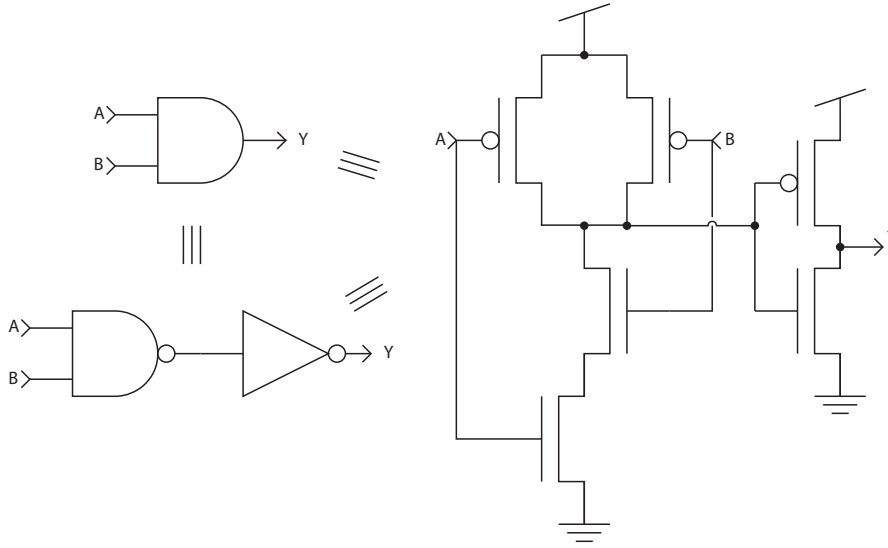


Figure A.21: Solution task A.3.3

Solution A.3.2

$$\overline{a \vee b \vee c} = \bar{a} \wedge \bar{b} \wedge \bar{c} \quad (\text{A.70})$$

$$\overline{a \vee b \wedge c} = \bar{a} \wedge (\bar{b} \vee \bar{c}) \quad (\text{A.71})$$

$$\overline{(a \vee b) \wedge c} = \bar{a} \wedge \bar{b} \vee \bar{c} \quad (\text{A.72})$$

$$\bar{a} \vee \bar{b} \vee \bar{c} = \overline{a \wedge b \wedge c} \quad (\text{A.73})$$

$$\bar{a} \vee \bar{b} \wedge \bar{c} = \overline{a \wedge (b \vee c)} \quad (\text{A.74})$$

$$(\bar{a} \vee \bar{b}) \wedge \bar{c} = \overline{a \wedge b \vee c} \quad (\text{A.75})$$

Solution A.3.3

See figure A.21! The task asks for the transistor level solution, so that is the important one to submit, if this were a exam task. The others er more high level representations for illustration. They are equivalent only if it is understood that they represent complementary PUN/PDN style logic.

Solution A.3.4

PUN5

Solution A.3.5

From fastest to slowest transition: (3,4,2,1)

Solution A.3.6

From fastest to slowest transition: (1,4,2,3)

Solution A.3.7

Port 1 and 6

Solution A.3.8

The tasks states to assume that delay is governed purely by the input capacitances of the next gates a gate is connected to. Also an inverter uses a 2.5 times wider pFET than nFET. Thus relative input capacitances for balanced gates are (assuming the gate capacitance of the inverter nFET is 1 and thus the inverter's pFET 2.5):

inverter: $1 + 2.5 = 3.5$

2 input NAND: $2 + 2.5 = 4.5$

2 input NOR: $1 + 5 = 6$

3 input NAND $3 + 2.5 = 5.5$

resulting worst case (meaning all gates switch) relative gate delays in the marked nodes are:

$a = 4.5$

$b = 6$

$c = a + (5.5 + 4.5) = 14.5$

$d = 4.5 + 2 \cdot 4.5 = 13.5$

$e = c + 3.5 = 18$

$f = b + (4.5 + 5.5) + 3.5 = 19.5$

It's not really necessary to solve at this level of detail: It's enough to realize that the 2 input nor has more input capacitance than the nand so $a < b$ and then that the three input nand has a bigger input capacitance than the 2 input nand, so $d < c$. Then the unmarked node below d has bigger delay than c (they have identical capacitance but the node below d suffers from the delayed b) and therefore $e < f$

So finally:

$a < b < d < c < e < f$

Solution A.3.9

See figure A.22. It uses $\sim C$ and $\sim \phi$ as inversion instead of $\overline{C}$ and $\overline{\phi}$

Solution A.3.10

See figure A.23

Solution A.3.11

Truth table

J K D

0 0 Q

0 1 0

1 0 1

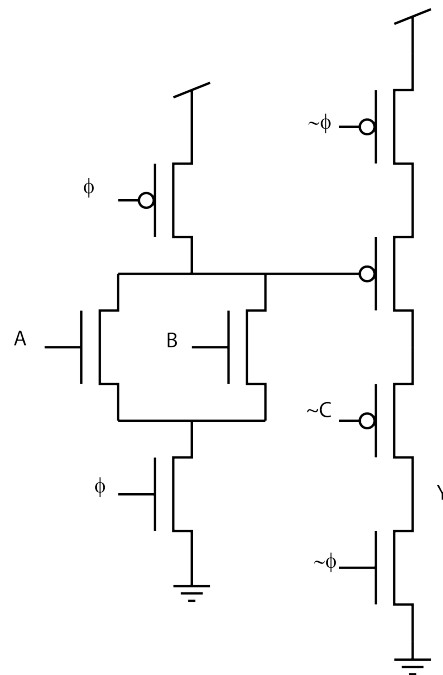


Figure A.22: Solution task A.3.9

1 1 notQ

More explicit

J	K	Q	D
0	0	0	0
0	0	1	1
0	1	X	0
1	0	X	1
1	1	0	1
1	1	1	0

Simplest implementation with 2x3-input AND gates and 1x2-input AND gate in the three cases where D is 1

Solution A.3.12

See figure A.24. It uses $\sim M$ and $\sim H$ etc. as inversion instead of $\overline{M}$ and $\overline{H}$, etc. Note that this is but one possible solution! Other equivalent implementations are possible.

Solution A.3.13

See the sketch in figure A.25. A straight forward implementation is a counter that counts from 1 to 15 and combinational logic that deduces the output states

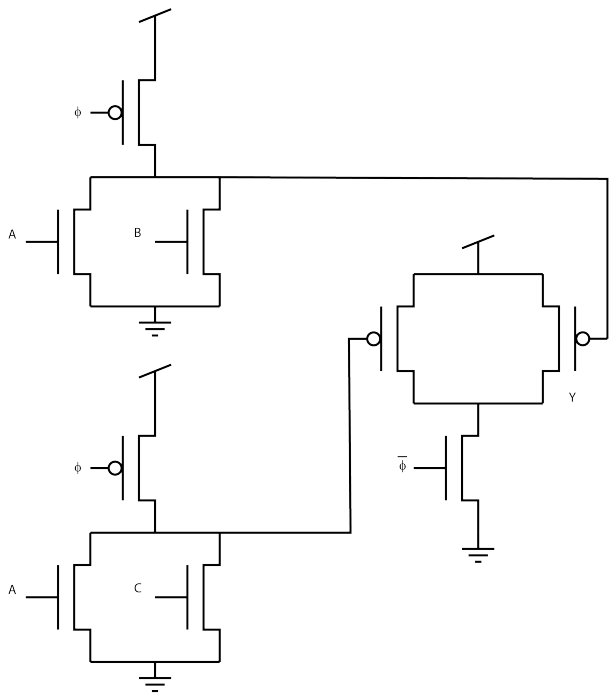


Figure A.23: NP domino logic solution for A.3.10.

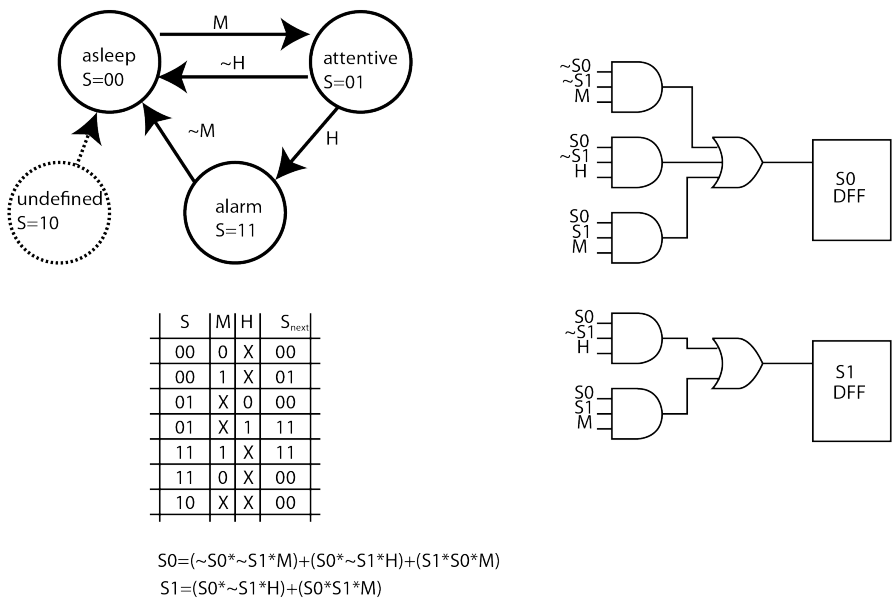


Figure A.24: Solution task A.3.12

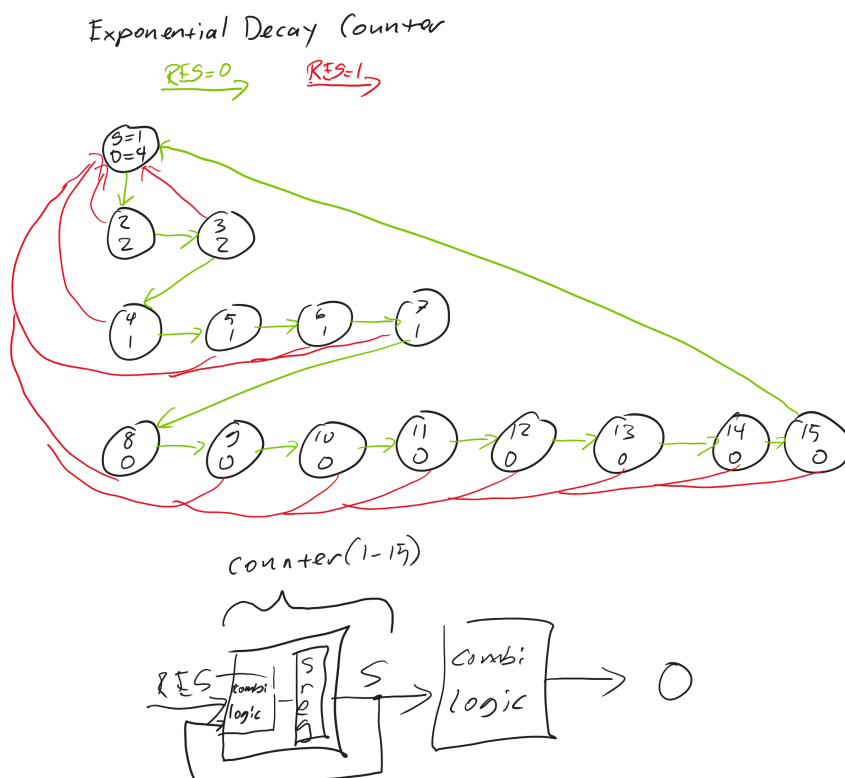


Figure A.25: Exponential Decay counter state transition graph solving A.3.13.

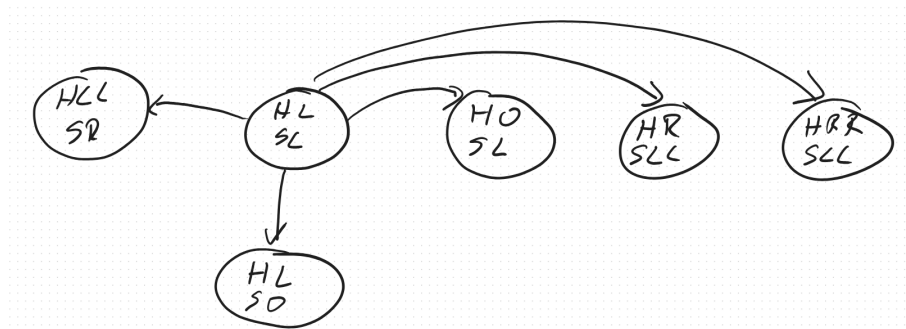


Figure A.26: Solution task A.3.14

from the counter values.

Solution A.3.14

See figure A.26. Note that since there are 5 possible sensor inputs there needs to be state transitions for all 5 of these (some may return to the same state or multiple inputs can also lead to the same state). The drawing omits the state transition label for brevity: it's redundant since the state stores the last sensor input, so the heading *leading* to a state is also the heading *stored* in that state. In that sense, each of the 5 transitions has to end up in one of the 5 columns of the state bobbles as organized in the figure A.18 given in the task.

Solution A.3.15

```

function [tp2,tp3,tp4]=PropagationDelay(tp1,m,k)

n=5;

tp2=m*tp1
tp3=5*m^(1/5)*tp1
%tp4=4*m^(1/5)*tp1+ k*m^(1/5)*tp1
tp4=(4+k)*m^(1/5)*tp1

end
  
```

Solution A.3.16

See figure A.27.

Solution A.3.17

See figure A.28. Other correct solutions are possible.

Solution A.4.15

```

function [ViMax, ViMin, Vi1Max, Vi1Min, Vi2Max, Vi2Min, VoMax, ...
  
```

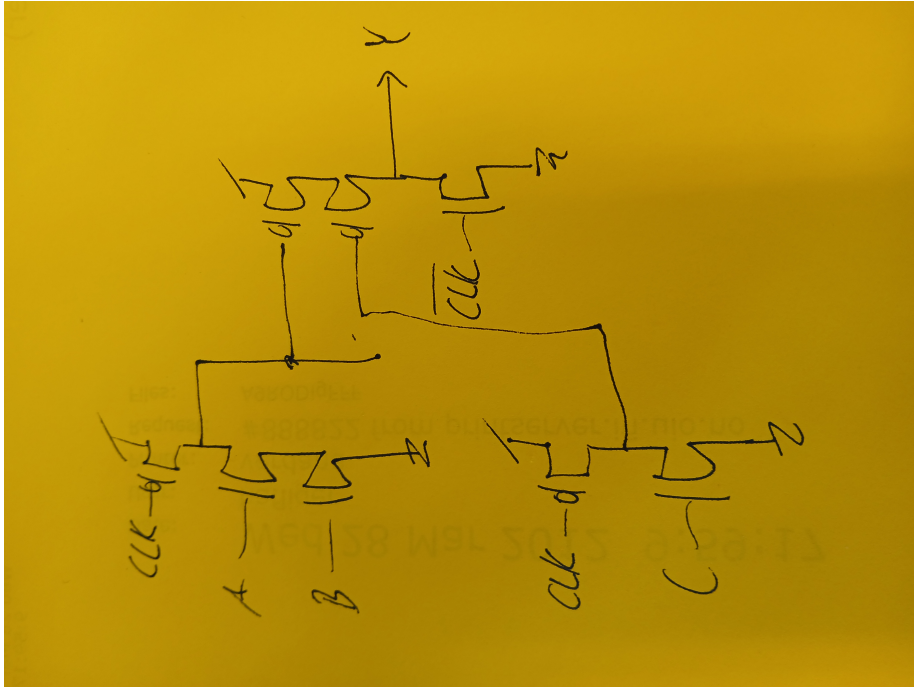


Figure A.27: Solution task A.3.16

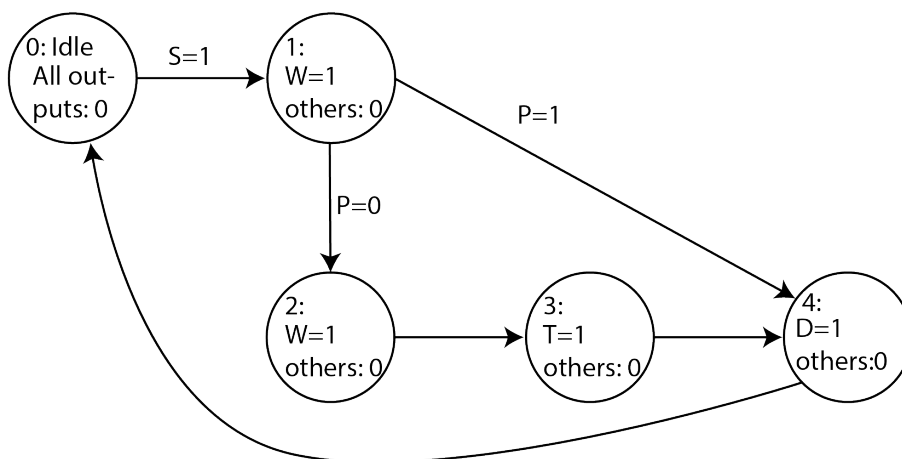


Figure A.28: Solution task A.3.17

parameter	180nm		45nm	
	N	P	N	P
μC_{OX} $\left[\frac{\mu A}{V^2}\right]$	280	80	290	90
V_t [V]	0.45	0.45	0.45	0.45
Vdd-Vss [V]	1.8		1.2	
λL $\left[\frac{\mu m}{V}\right]$	0.08	0.08	0.11	0.15
n	1.6	1.7	1.8	1.8

Table A.1: Process parameter examples

```

VoMin, Vo1Max, Vo1Min, Vo2Max, Vo2Min]=CurrentMirrorVoltages(Vov,Vt, Vdd)

ViMax=3*Vt+3*Vov;
ViMin=ViMax;

Vi2Max=2*Vt+2*Vov;
Vi2Min=Vi2Max;

Vi1Max=Vt+Vov;
Vi1Min=Vi1Max;

Vo1Max=Vi1Max;
Vo1Min=Vo1Max;

Vo2Max=Vi2Max;
Vo2Min=Vo2Max;

VoMin=Vo2Max+Vov
VoMax=Vdd

end

```

A.4 Analog Circuit Basics

A.4.1

Let us assume some typical process parameters for a 180nm and a 45nm technology in table A.1. Consider the intrinsic gain of an nFET in 45nm and 180nm technology according to this table. What are the intrinsic gains A45 and A180 of a 45nm and 180nm nFET respectively, if both are biased at $40\mu A$, have a length L45 and L180 that is two times their respective minimum length and are three times as wide as long. Assume strong inversion and saturation! For the purpose of deriving g_m and r_{ds} you may assume that $I_D \approx I'_D = 40\mu A$.

Note in this table λL is a process specific constant.

Be careful about using all units correctly: one option is to get rid of all the micro and kilo etc. in the units by writing *1e-6 and *1e3 respectively. Submit the result as a MATLAB script in file IntrinsicGain.m. The final result after executing the script needs to be in variables A45 and A180 respectively. To

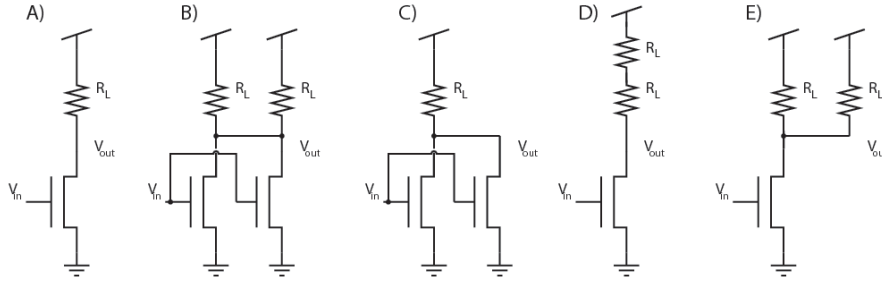


Figure A.29: CS stages to be ordered by their gain with same input bias V_I

make your file readable, use '45' and '65' in your variable names if they are process specific, e.g. W45 and W180. Use variable names that are similar to the ones used in the equations in the book. Please upload MATLAB script in file IntrinsicGain.m with the final result in variables A45 and A180!

A.4.2

For a nFET with $k_n = 400 \frac{\mu A}{V^2}$ (pay attention to the 'micro' in the unit!) and threshold voltage $V_{tn} = 0.5V$ biased at a current of $15\mu A$, and a $C_{GS} = 10fF$ and $C_{GD} = 1fF$, compute the current unity gain frequency ω_T in the MATLAB script fT.m, i.e. it shall not be a matlab function with a function header but simply a series of matlab commands at the end of which the result shall be in variable f_T.

A.4.3

Please order these Common source stages according to their small signal DC gain A_{DC} . Assume that all transistors have the same dimensions W and L , and are biased in saturation with the same overdrive voltage V_{OV} , i.e. the same input DC-offset V_I . Also assume that $R_L = r_{ds}$, where r_{ds} is the small signal drain resistance of one of the nFETs in the active region. If two or more CS-stages should happen to have the same A_{DC} you can place them next to each-other in any order.

A.4.4

Given a CS stage (see illustration A.30) where $\mu C_{ox} = 300 \frac{\mu A}{V^2}$, $V_t = 600mV$, $n = 1$, $W = 1200nm$, $L = 600nm$, $V_S = 0V$, $V_G = 450mV$, $R_L = 210M\Omega$, $V_{dd} = 1.8V$, $A = 20 \frac{V}{V}$, derive g_m , r_{ds} , and the process parameter $\lambda * L$.

Assume that the transistor operates in saturation.

Submit your solution in a matlab script that assigns a numeric (!) value to variables 'gm', 'rds' and 'lambdaL' when it is executed, so with final statements of the form:

```
gm= ...
```

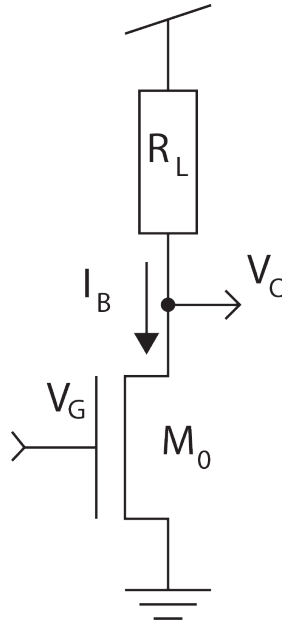


Figure A.30: Generic CS stage for task A.4.4

`rds=...`

`lambdaL=...`

as well as (optionally) statements leading up to that final result.

Hint: to derive parameter g_m and r_{ds} in dependency of I_D it is appropriate to use the value for I_D without channel length modulation effect.

Save it in file 'parameterLambdaL.m' and upload that file!

A.4.5

Imagine you have a current source that should provide a current to a load, e.g. a motor or a heating element, as illustrated by the circuit schematics 1) (fig. A.31). As you increase the current i_i , much of that current does not drive the load R_L but is 'wasted' through the source's internal resistance R_S . The Circuit 2) tries to improve that situation by interposing a common gate (CG) stage (can also be regarded as a cascode transistor) between the source and the load. Here this CG stage has the function of a current conveyor, increasing the percentage of the source current actually ending up driving the load, i.e. less current being wasted on the source's internal resistance. This becomes evident if you derive expressions (Do this!) for the small signal (!) current gains for the two situations, $A_{i1} = \frac{i_{o1}}{i_i}$, $A_{i2} = \frac{i_{o2}}{i_i}$ as functions of the four variables R_S, R_L, g_m, r_{ds} .

Submit a MATLAB function as file `CurrentConveyor.m` with function header:
`function (Ai1, Ai2)=CurrentConveyor(RS,RL,gm,rds)`

That computes A_{i1} and A_{i2} from the variables R_S, R_L, g_m, r_{ds} .

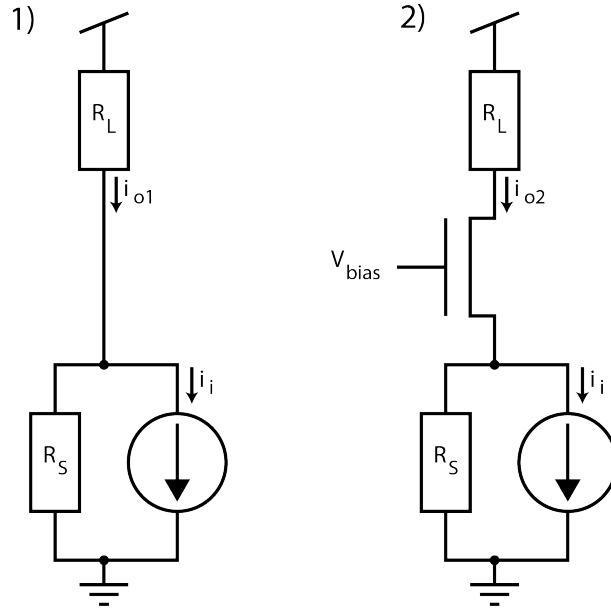


Figure A.31: Figure for task A.4.5: Using a cascode as a Current conveyor

A.4.6

Find the correct load resistance R_L , the total transconductance G_M , the output resistance R_O looking into the transistor source (so not the total resistance R_L^* seen at the output node) and the two poles p_1, p_2 and the zero z_1 of a source follower (SF), aka common drain stage (see illustration A.32!), given the following parameters:

$$\begin{aligned}
 \mu C_{OX} &= 320 \frac{\mu\text{A}}{\text{V}^2} \\
 \frac{W}{L} &= 1 \\
 n &= 1 \\
 \lambda &= 0.05 \\
 V_t &= 0.3\text{V} \\
 V_{dd} &= 1.8\text{V} \\
 V_I &= 1.2\text{V} \\
 U_T &= 26\text{mV} \\
 I_D &= 50\text{nA} \\
 C_{GS} &= 10\text{fF} \\
 C_{GD} &= 1\text{fF} \\
 C_L &= 100\text{fF} \\
 R_S &= 100\text{k}\Omega
 \end{aligned}$$

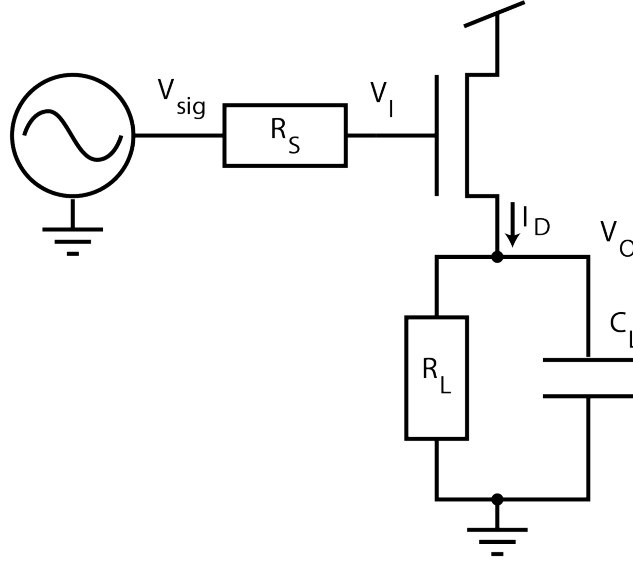


Figure A.32: Source follower for task A.4.6

C_{BS}, C_{BD} are negligibly small.

You may assume that the transistor is in weak inversion and that the poles will be real. The solution for quadratic equations may come in handy, i.e. $x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ are the two solutions for $0 = ax^2 + bx + c$. Use the exact solution for a general transconductance amplifier from section 9.4.1 (!) applied to a source follower. Also check the new table 9.2 in the compendium for the appropriate capacitors to be used for a CD/SF.

Submit the solution as a MATLAB script in file 'SF.m' which needs to assign solutions to variables RL, GM, RO, p1, p2, and z1 (for respectively R_L , G_M , R_O , p_1 , p_2 and z_1)

A.4.7

An nFET differential pair is loaded with a pFET current mirror and an ideal current source provides a bias current I_B of $100\mu\text{A}$ (i.e. this is a current mirror loaded differential amplifier, see illustration A.33). Compute its differential small signal gain $A = \frac{v_o}{\Delta v_i}$ if $k_n = 500 \frac{\mu\text{A}}{\text{V}^2}$ and $k_p = 300 \frac{\mu\text{A}}{\text{V}^2}$ and the Early voltage $V_A = \frac{1}{\lambda} = 20\text{V}$ for both type of transistors.

A.4.8

The illustration A.34 shows the small signal model of a CS gain stage. Given that $g_m = 1 \frac{\text{mA}}{\text{V}}$, $r_{ds} || R_L = 30\text{k}\Omega$, $R_{sig} = 4.8\text{k}\Omega$, $C_{gs} = 100\text{fF}$ and the -3dB cut off is $\omega_{-3\text{dB}} = 100\text{Mrad}$, what is the DC gain A_{DC} and C_{gd} ?

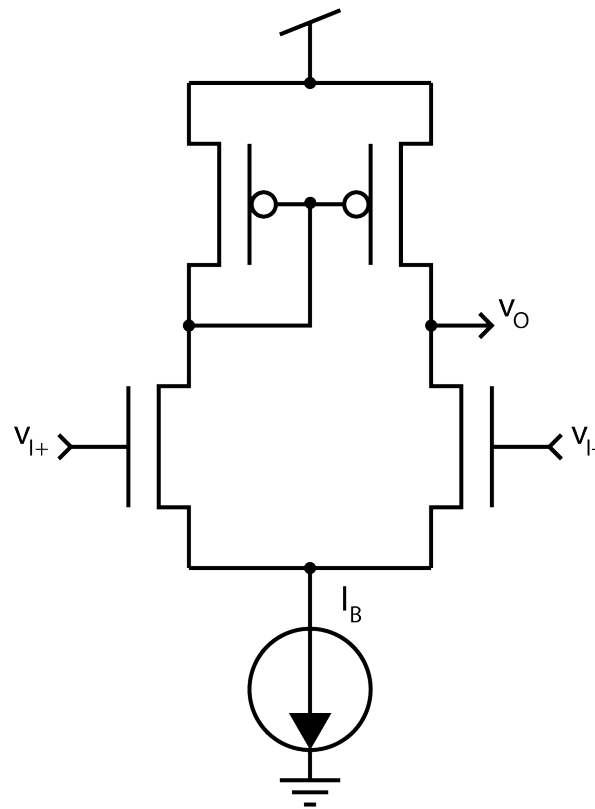


Figure A.33: Schematic of a current mirror loaded differential pair with ideal bias current source

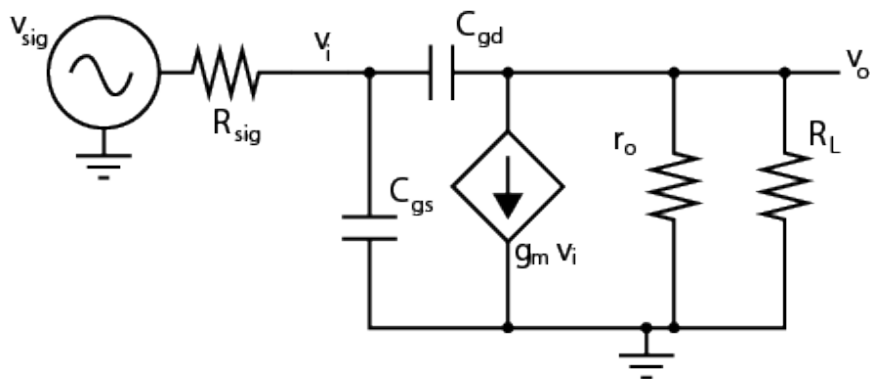


Figure A.34: CS small signal schematics to compute Miller capacitance

A.4.9

Can you list effects/properties that will affect the output current I_O of a current mirror and cause deviation of the output current from the input current I_I (intentional or unintentional). Please make your answers a list of just 1-3 sentences per effect/property.

A.4.10

(*At the exam this question was somewhat difficult, because students did not really know about using vectors in MATLAB, so in future iterations I will ask for 5 individual results Q1, Q2, Q3, Q4, Q5 for Rsig1, Rsig2, Rsig3, Rsig4, Rsig4*)

Find the Q-factor of a source follower! Use $C_{GS} = 2\text{pF}$, $C_{GD} = 0.1\text{pF}$, $g_m = 5 \frac{\text{mA}}{\text{V}}$, $C_L = 1\text{pF}$, $R'_L = 30\text{k}\Omega$. ($R'_L := r_{ds} || R_L$ and is used like that in an equation for Q of a SF in the Sedra & Smith book [2]. This compendium uses a variable R_L^* that is differently defined! You can derive it from R'_L and other given variables in this task here.) Any other capacitors and resistors are considered negligible. Compute 5 results for 5 values of Rsig=[200 500 5000 50000 125000] Ω what is the Q-factor Q? Upload MATLAB script Qfactor.m which assigns final result to variable Q where Q is a vector with 5 elements corresponding to Rsig=[200 500 5000 50000 125000]!

A.4.11

A current mirror loaded differential pair without an external load resistance (i.e. $R_L = \infty$) has an open circuit differential gain of $A = 18 \frac{\text{V}}{\text{V}}$. When an external load of $R_L = 30\text{k}\Omega$ is connected, the gain is reduced to $A = 12 \frac{\text{V}}{\text{V}}$. What is the circuit's short circuit transconductance $G_M = ? \frac{\text{A}}{\text{V}}$ and output resistance $R_O = ? \Omega$? Submit the answer as a MATLAB script in file Transamp.m. The script shall assign the answer for G_M and R_O to MATLAB variables 'GM' and 'RO' respectively.

A.4.12

Using the process parameters from table A.1 for the 45nm process, what is the gain of a transconductance amplifier if constructed with transistors that all have the same width $W = 1\mu\text{m}$ and length $W = 1\mu\text{m}$? Attention: make it a p-type differential pair and thus, use nFETs as the current mirror. Use a 10nA bias current. You may assume that the bulks are all tied to the sources (as a consequence there is no back gate effect, i.e. you may assume $n = 1$. Beware that if the bulks of the two transistors in the differential pair would be connected to the supply rails instead, then n could not be ignored and the results would be influenced.)

Given a $C_{ox} = \frac{25\text{fF}}{\mu\text{m}^2}$ and a L_{OL} in the drain area of 1nm, what are the expected 2 poles and the one zero of the transfer function. Furthermore we assume $C_{DB} = 0.05\text{fF}$ and a load capacitance $C_L = 0.1\text{fF}$

A.4.13

For a transconductance amplifier (aka current mirror loaded differential pair), deduce the range for the common mode input and the range of output voltage such that all transistors are in saturation for a zero differential input. To make this of the region of operation use parameters V_{satB} and V_{GSB} for the saturation voltage (i.e. the minimum V_{DS}) and the gate to source voltage of the bias transistor, V_{satD} and V_{GSD} for the differential pair transistors, and V_{satM} and V_{GSM} for the current mirror voltages. You may assume that all V_{GS} are bigger than their respective V_{sat} . For the output voltage range you may also use the common mode input voltage V_{CM}

Once you have the result expressed with these parameters, express them also for the strong inversion case with the threshold voltages V_{tn} , V_{tp} , as well as the overdrive voltages V_{OVn} (for the diff pair transistors) and V_{OVp} (for the current mirror). Note that there is a relationship between V_{OVn} and the overdrive voltage of the bias transistor, assuming it has the same dimensions as the differential pair transistors, so you can use that and do not need another parameter for the bias transistor overdrive voltage.

A.4.14

The illustration shows a so called Willson current mirror. It works just like a simple two transistor current mirror, but achieves a much better output resistance, making it a better current source. So just like for a simple current mirror, if all transistors are in saturation and the channel shortening effect is insignificant, then the output current I_O matches the input current I_I exactly.

Make a matlab function that describes the condition for the voltages at the input V_I and output V_O such that the output current I_O is the same as the input current I_I (i.e. $I_O = I_I$) while all three transistors are in saturation. Please neglect channel length modulation and back gate effects (i.e. assume that $n = 1$ and $\lambda = 0$).

a) V_I and V_O may either need to be an exact value or they may need to be bigger than a minimum value. To indicate whether the following results for these two voltages indicate an exact or minimal value, assign a '1' or a '0' to variables 'VOexactNOTmin' and 'VIexactNOTmin', where '1' means that the value that your function returns is an exact value and '0' means that it is a minimal value.

b) Derive expressions for the two voltages for the general case, given a saturation voltage V_{sat} (variable name 'Vsat' in Matlab) that is the minimum voltage across the drain and source terminals for a transistor to be in saturation, and V_{GS} (variable name 'Vgs' in Matlab) which is the gate to source voltage required for one transistor in saturation to conduct $I_O = I_I$. Add thus two expressions 'VO=...' and 'VI=...' into the function where you may only use 'Vgs' and 'Vsat' on the righthand side and possibly natural constants (as constant numbers or variables that are assigned a constant number).

Use this function header in matlab file 'WillsonVoltages.m' that you then upload:

```
function [VO, VI, VOexactNOTmin, VIexactNOTmin]=WillsonVoltages(Vgs,Vsat)
```

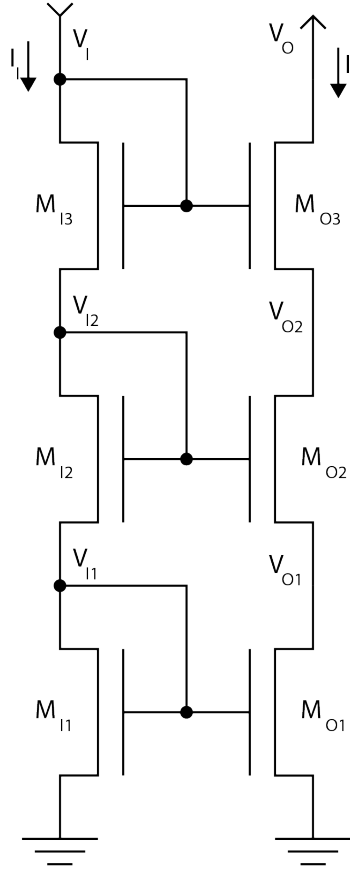


Figure A.35: Double Cascode Current Mirror (not necessarily a good idea in practice).

A.4.15

The illustration A.35 shows a current mirror with two added casacode transistors, i.e. three nFETs in series on both the input and output branch. Deduce the maximum and minimum voltages for each of the nodes that keep all transistors in saturation (!!!) and conducting the same current when given the overdrive voltage V_{ov} and threshold voltage V_t appropriate for the input current. All transistors are identical. Assume strong inversion! Report the voltages and voltage ranges as a matlab function in file `CurrentMirrorVoltages.m` with function header

```
function [ViMax, ViMin, Vi1Max, Vi1Min, Vi2Max, Vi2Min, VoMax, VoMin,
Vo1Max, Vo1Min, Vo2Max, Vo2Min]=CurrentMirrorVoltages(Vov,Vt, Vdd)
```

Ignore any channel length modulation or back gate effect, i.e. $n=1$ and $\lambda = 0$. If the maximum for any of these voltages is not limited, simply assign V_{dd} as the upper limit for it, and if the minimum is not limited, assign zero to it. If any of these voltages has an exact value instead of a range, simply use the same expression for both the maximum and minimum value.

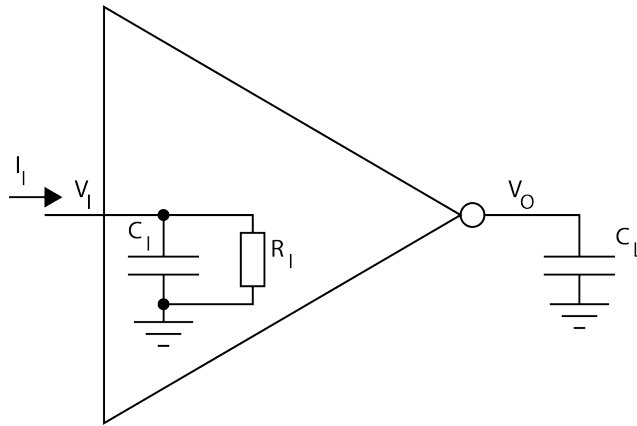


Figure A.36: Equivalent schematic for input impedance.

A.4.16

Suppose you buy an inverter electronic circuit as a black box with supply pins and a pin for an input and output. So you do not exactly know, how the inverter is implemented inside that black box. It might not even be a CMOS circuit, but functionally for DC inputs it does what it is supposed to do, i.e. it inverts a voltage input.

In order to know what signal you need to provide from another circuit to its input, you want to characterize its input impedance. You find out that a simple input capacitance alone is insufficient to model its input impedance. Instead, it looks like it behaves like there is an input resistance in parallel with an input capacitance, as indicated in the schematic of the figure A.36.

To find values for the input capacitance C_I as well as for the input resistance R_I you apply an input step current I_I and monitor the input voltage V_I and output voltage V_O . The supply rails are specified as 1.2V Vdd and 0V for Gnd and are connected.

The resulting signal traces are shown in figure A.37. The graphs indicate a few points and lines with values. Some of them will be helpful to solve the task, others might just be distractors.

Find the values for R_I (R_i in MATLAB-script) and C_I (C_i in MATLAB-script) and report them in a MATLAB script as variables R_i and C_i . The script shall be submitted in file:

"inputImpedance.m"

and contain statements that assign the correct numerical values to the variables

$C_i = ?$

$R_i = ?$

A.4.17

The figure shows the voltage traces from an experiment with a differential amplifier. V_{in+} and V_{in-} are the differential inputs and V_{out} is the single ended

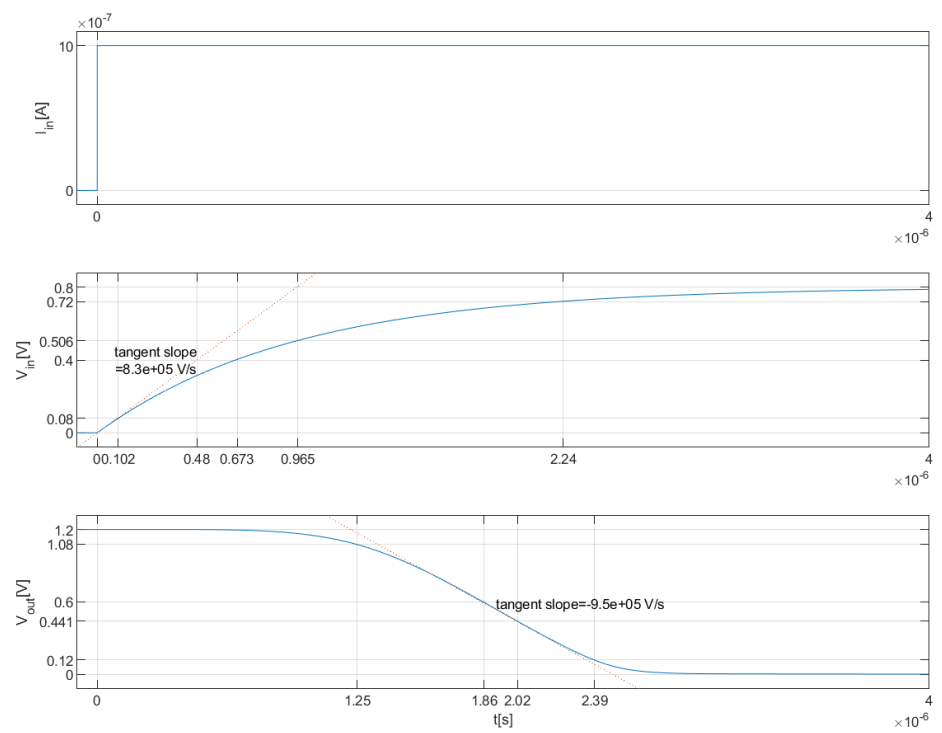


Figure A.37: Measurement to deduce input impedance.

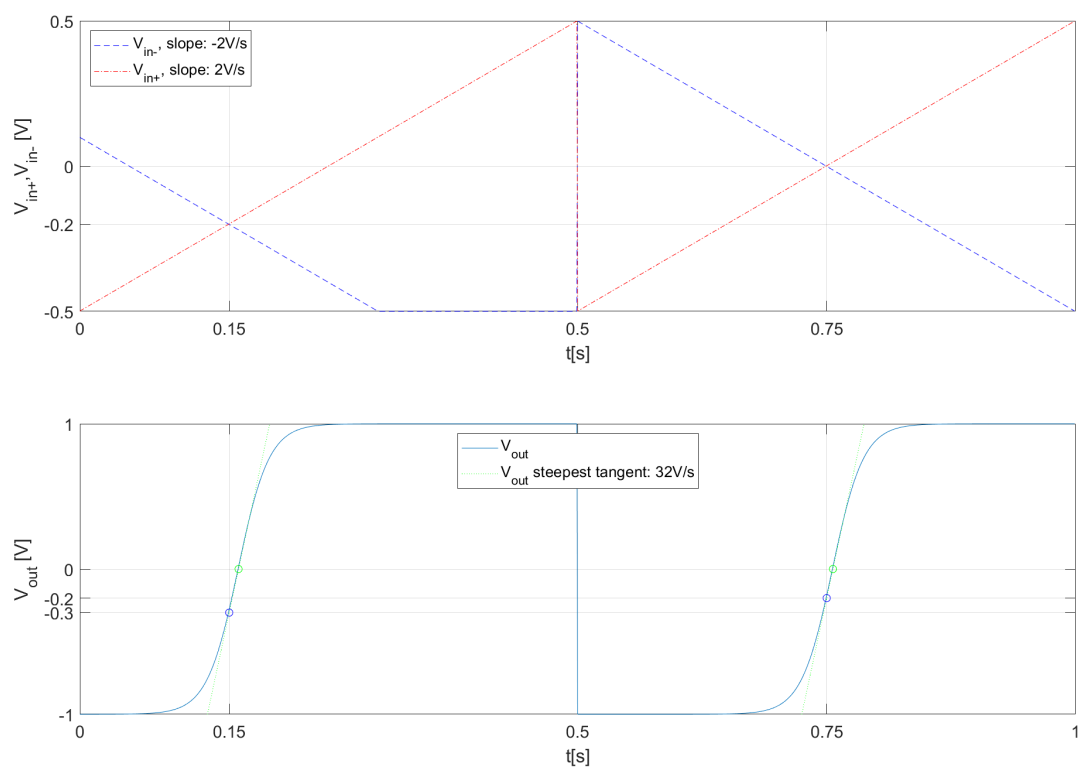


Figure A.38: Measurement to deduce CMRR.

output. The signals are slow enough to be well inside the pass band of the amplifier so there are no frequency effects and the observed behaviour is the amplifier's low frequency/DC behaviour.

From the graphs and the indicated time and voltage values, derive the small signal gain A , the common mode gain A_{cm} , and the common mode rejection ratio CMRR in decibel (!).

Submit the result as a MATLAB-script in file "CMRR_script.m" that assigns the correct numerical values to variables:

$A=?$

$A_{cm}=?$

CMRR=?

Use at least 3 significant digits!

Solutions section A.4

Solution A.4.1

```
function [A] = IntrinsicGain( Id, L, uCox, W, LamL)

%gm=Vov*kn;
%Id=0.5*kn*Vov^2
gm=sqrt(2*Id*uCox*W/L)
%lambda=lamL/L
ro=L/LamL/Id

%A=sqrt(2*Id*uCox*W/L)*L/lamL/Id
A=sqrt(2*L/Id*uCox*W)/LamL
end
```

Then you may execute:

```
Id=40e-6;
L45=45*2e-9;
L180=180*2e-9;
W45=3*L45;
W180=3*L180;
uCox45=290e-6;
uCox180=280e-6;
LamL45=0.11e-6;
LamL180=0.08e-6;

[A45] = IntrinsicGain( Id, L45, uCox45, W45, LamL45)

[A180] = IntrinsicGain( Id, L180, uCox180, W180, LamL180)

A45 =

5.3963
```

A180 =

29.1633

Solution A.4.2

```
d=15e-6
kn=400e-6
Cgs=10e-15;
Cgd=1e-15;

%gm=Vov*kn;
%Id=0.5*kn*Vov^2
gm= sqrt(2*Id*kn)

f_T=gm/(Cgs+Cgd)/2/pi

f_T =

1.5850e+09
```

Solution A.4.3

The gain is given through the total transconductance G_M multiplied with the total output resistance R_O . G_M is multiplied with the total transistors in parallel, and R_O is reduced with the total transistors *and* load resistors in parallel and increases if resistors are in series. Thus:

$$\begin{aligned}
 A_A &= \frac{1}{2} g_m r_{ds} \\
 A_B &= \frac{1}{2} g_m r_{ds} \\
 A_C &= \frac{2}{3} g_m r_{ds} \\
 A_D &= \frac{2}{3} g_m r_{ds} \\
 A_E &= \frac{1}{3} g_m r_{ds}
 \end{aligned} \tag{A.76}$$

$$A_C = A_D > A_A = A_B > A_E \tag{A.77}$$

Solution A.4.4

A possible Matlab script:

```
UT=26e-3
Vt=0.6
```

```

RL=210e6
A=20
Vg=0.45
W=1200e-9
L=600e-9
uCox=300e-6
k=uCox*W/L
Is=2*k*UT^2
Id=Is*exp((Vg-Vt)/UT)
gm=Id/UT
%gm = 9.7411e-08
RsL=A/gm

%RsL= rds*RL/(RL+rds)
%RsL*rds + RsL*RL=rds*RL
%RsL*RL = rds*(RL -RsL)
rds=RsL*RL/(RL-RsL)
%rds = 9.2028e+09

%from: rds=L/(lambdaL*Id)
lambdaL=L/(rds*Id)

% lambdaL=2.5742e-08
% lambdaL/L=0.0429

```

Solution A.4.5

```

function [Ai1,Ai2]=CurrentConveyor(RS,RL,gm,rds)

Ai1=RS/(RS+RL);
Ai2=RS/(RS+1/gm+RL/(gm*rds));
%Ai2=RS/(RS+(rds+RL)/(1+gm*rds)); %more precisely

```

Solution A.4.6

G_M and R_O can directly be deduced from I_D and the given device parameters. Since the drain current and gate voltage (input point of operation) V_I is given, one can then deduce the output point of operation V_O and from I_D and V_O the appropriate R_L .

From there one can use the general transconductance amplifier model in chapter 5 for p_1, p_2, z_1 , as indicated in the task text. See equation (5.26) in version beta 0.15 of the compendium. From that equation one needs to solve a quadratic equation to get the exact poles.

```

uCox=320e-6 % A/V^2
WL=1
n=1
lambda=0.05
Vt=0.3
Vdd=1.8

```

```

VI=1.2
UT=26e-3
k=WL*uCox
Is=2*n*k*UT^2
%Is = 4.3264e-07
Id=50e-9

%Id=Is*exp((VI-Vt-n*V0)/n/UT)
%log(Id)-log(Is)=(VI-Vt-n*V0)/n/UT
V0=((log(Is)-log(Id))*n*UT-Vt+VI)/n
%V0 = 0.9561
RL=V0/Id
%RL=1.9122e+07

gm=Id/UT
%gm =1.9231e-06
rds=1/lambda/Id
%rds = 400000000

Rsig=100e3
Cgs=10e-15
Cgd=1e-15
CL=100e-15
R0=rds/(1+gm*rds)
%R0 = 5.1932e+05
CsL=CL
Cfb=Cgs
CI=Cgd
RsL=R0*RL/(R0+RL)
RsI=Rsig
GMA=-gm
A=RsI*RsL*((Cfb+CsL)*(Cfb+CI)-Cfb^2)
B=RsL*(Cfb+CsL)+RsI*(CI+Cfb*(1+GMA*RsL))

p1=(-B+sqrt(B^2-4*A))/2/A
%p1 =-1.8276e+07

p2=(-B-sqrt(B^2-4*A))/2/A
%p2 =-9.7499e+08

z1=GMA/Cfb
%z1 = -1.9231e+08
GM=GMA;

```

Solution A.4.7

Once more we assume strong inversion (with μA biasing it is most likely the case) which needs verification at the end.

Q =

0.4886 0.7324 1.3005 0.7638 0.5123

%Philipp:

%%%%%%%%%

%RLstar=?

%RLdash=rds*RL/(rds+RL)

RLstar=RLdash*1/gm/(1/gm+RLdash) % in Philipp it's (rds || RL || 1/gm)

A=RLstar*Rsig * ((Cgs+CL)*(Cgs+Cgd)- Cgs^2)

B=RLstar*(Cgs+CL)+Rsig*(Cgd+Cgs*(1-gm*RLstar))

Q=sqrt(A)./B

Q =

0.4886 0.7324 1.3005 0.7638 0.5123

Solution A.4.11

%R0*RL/(R0+RL)=2/3*R0

%1/RL+1/R0=3/2*R0

RL=30e3;

R0=0.5*RL;

%R0=15e3

%Gm*R0=18

Gm=18/R0

%Gm=1.2e-3

Solution A.4.12

Are we in strong or weak inversion?

$$I_S = 2nkU_T^2 = 2 * 1.8 * 90e - 6 * 1/1 * 26e - 6 = 84nA \quad (A.81)$$

So with a bias current of 10nA resulting in 5nA in the branches we are in sub-threshold.

Thus we get the small signal parameters for the branches (both nFET and pFET)

$$g_m = \frac{I_D}{U_T} = 5e - 9/26e - 3 = 192 \frac{nA}{V} \quad (A.82)$$

$$\begin{aligned}
r_{dsp} &= \frac{L}{L\lambda I_D} = 1e - 7/5e - 9/0.15e - 6 = 133\text{M}\Omega \\
r_{dsn} &= \frac{L}{L\lambda I_D} = 1e - 7/5e - 9/0.11e - 6 = 182\text{M}\Omega
\end{aligned} \tag{A.83}$$

So we get a gain of

$$A = g_m(r_{dsn} || r_{dsp}) = 192e - 9 * (133e6 * 182e6 / (133e6 + 182e6)) = 14.75 \frac{\text{V}}{\text{V}} \tag{A.84}$$

Then we get the capacitances:

$$\begin{aligned}
C_{GD} &= L_{OL} * W * C_{ox} = 1e - 9 * 100e - 9 * 25e - 15/1e - 12 = 25e - 18 \\
C_{GS} &= (L_{OL} + 2/3 * L) * W * C_{ox} = (1e - 9 + 2/3 * 100e - 9) * 100e - 9 * 25e - 15/1e - 12 = 1.69e - 16 \\
C_{DB} &= 0.05\text{fF} \\
C_L &= 0.1\text{fF} \\
C_L^* &= 2C_{GD} + 2C_{DB} + C_L = 2 * 25e - 18 + 2 * 0.05e - 15 + 0.1e - 15 = 0.25\text{fF} \\
C_D &= 2C_{GD} + 2C_{DB} + 2C_{GS} = 2 * 25e - 18 + 2 * 0.05e - 15 + 2 * 1.69e - 16 = 0.49\text{fF}
\end{aligned} \tag{A.85}$$

Thus we have a pole frequency of

$$\begin{aligned}
\omega_{p1} &= \frac{1}{C_L^*(r_{dsn} || r_{dsp})} = 1/(0.25e - 15 * (133e6 * 182e6 / (133e6 + 182e6))) = 52\text{MHz} \\
\omega_{p2} &= \frac{g_{mp}}{C_D} = 192e - 9/0.49e - 15 = 392\text{MHz} \\
\omega_{z1} &= \frac{2g_{mp}}{C_D} = 784\text{MHz}
\end{aligned} \tag{A.86}$$

Solution A.4.13

$$\begin{aligned}
\min(V_{CM}) &= V_{satB} + V_{GSD} \\
\max(V_{CM}) &= \text{Vdd} - V_{SGM} + V_{GSD} \\
\min(V_O) &= V_{CM} - V_{GSD} + V_{satD} \\
\max(V_O) &= \text{Vdd} - V_{satM}
\end{aligned}$$

when substituting the right terms for weak and strong inversion it is noteworthy that there are different currents going through the bias transistor and the diff-pair transistors and current mirror: so in theory it might happen that they are not all in the *same* region of operation. But let's not bother with that here:

For weak inversion the equations become simply:

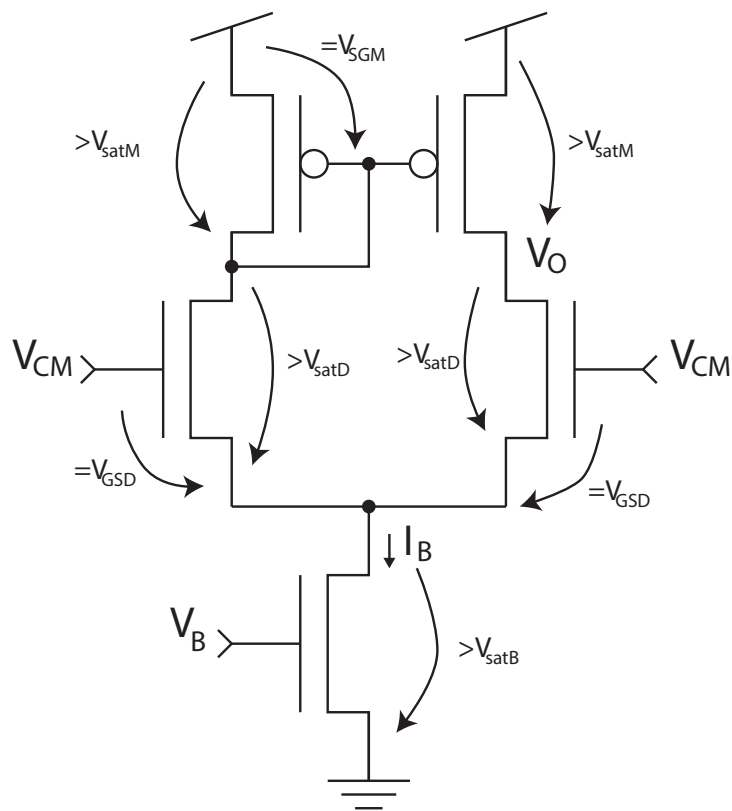


Figure A.39: Illustration for the solution of task A.4.13

$$\begin{aligned}
\min(V_{CM}) &= 4U_T + V_{GSD} \\
\max(V_{CM}) &= V_{dd} - V_{SGM} + V_{GSD} \\
\min(V_O) &= V_{CM} - V_{GSD} + 4U_T \\
\max(V_O) &= V_{dd} - 4U_T
\end{aligned}$$

For strong inversion:

$$\begin{aligned}
\min(V_{CM}) &= (\sqrt{2} + 1) V_{OVn} + V_{tn} \\
\max(V_{CM}) &= V_{dd} - V_{OVp} - V_{tp} + V_{OVn} + V_{tn} \\
\min(V_O) &= V_{CM} - V_{tn} \\
\max(V_O) &= V_{dd} - V_{OVp}
\end{aligned}$$

Solution A.4.14

Also see figure A.40! The matlab function:

```
function [VO, VI, VOexactNOTmin, VIexactNOTmin]=WillsonVoltages(Vgs,Vsat)

VIexactNOTmin=1;
VOexactNOTmin=0;
VO=Vgs+Vsat;
VI=Vgs+Vgs;
end
```

Solution A.4.15

On the input branch the voltage accross each diode connected transistor will be exactly the appropriate $V_{GS} = V_{tn} + V_{OV}$ for the strong inversion input current. On the output branch that is the same for the lower two voltages since M_{O3} and M_{O2} have the same gate voltages as the input transistors and need to have the same V_{GS} to conduct the same current, so their source voltages need to be the same. Thus, the only terminal that has a range of valid voltages is the drain of M_{O3} and V_O needs to be at least one $V_{sat} = V_{OV}$ (in strong inversion) above V_{O2} . So the correct matlab code for all this is:

```
function [ViMax, ViMin, Vi1Max, Vi1Min, Vi2Max, Vi2Min, VoMax, VoMin, Vo1Max, Vo1Min]

ViMax=3*Vt+3*Vov;
ViMin=ViMax;

Vi2Max=2*Vt+2*Vov;
Vi2Min=Vi2Max;

Vi1Max=Vt+Vov;
Vi1Min=Vi1Max;
```

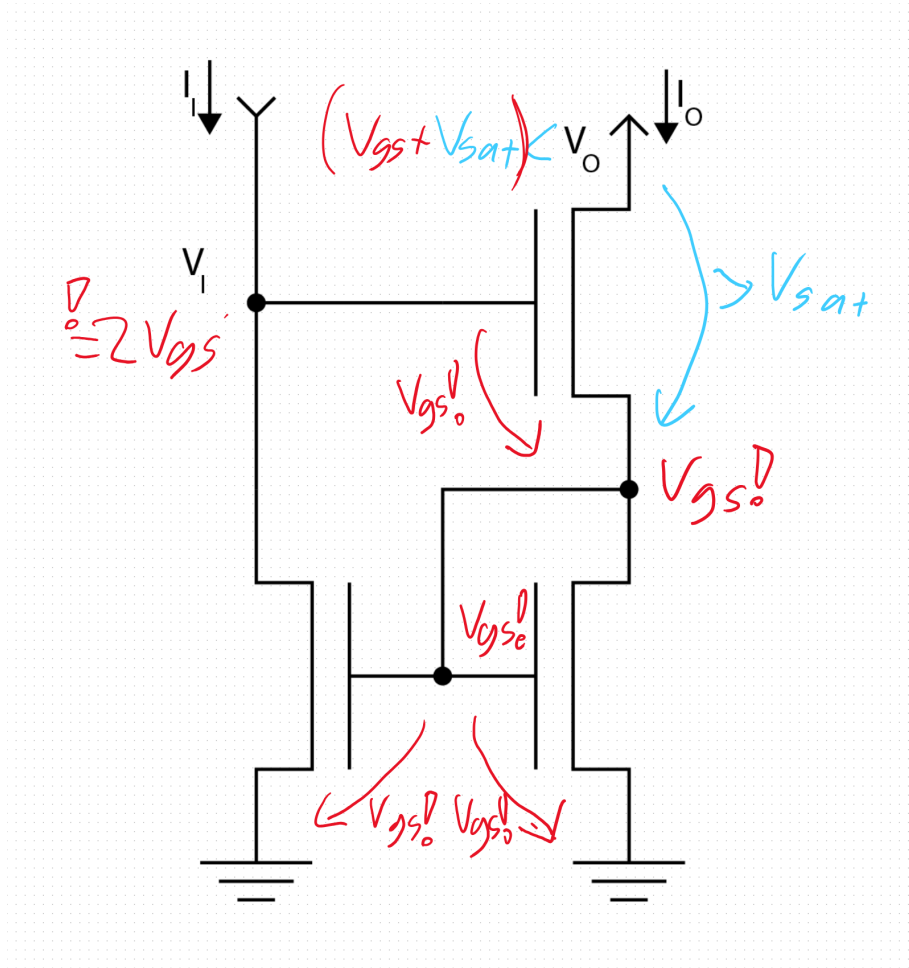


Figure A.40: Illustration for the solution for task A.4.14

```

Vo1Max=Vi1Max;
Vo1Min=Vo1Max;

Vo2Max=Vi2Max;
Vo2Min=Vo2Max;

VoMin=Vo2Max+Vov
VoMax=Vdd
end

```

Solution A.4.16

A first thing to notice is that the output trace V_O does in no way contribute to the deduction of the *input* impedance here. It's pure distraction. R_I is most easily deduced by observing where the input voltage settles and simply using $V_I = I_I R_I$ for this DC value of $V_I = 0.8V$. To deduce C_I directly one can observe the *instantaneous* reaction of $\frac{d}{dt} V_I = 830 \frac{kV}{s}$. At that moment the voltage across the resistor is 0V, so no current goes through it, i.e. all current flows onto the transistor and one can use $I_I = C \frac{d}{dt} V_I$. Alternatively one can use the result for R_I to get C_I and look at the time constant $\tau = R_I C_I$ of V_I , i.e. the time it takes until V_I arrives within *frac1e* of its final value, indicated in the graph as $0.965\mu s$.

So the Matlab code for this is:

```

%exact values used to produce the graph:
%C=1.2e-12;
%R=800e3;

%for instance deduced like this:
I=1e-6;

%The final voltage
V=0.8
Ri=V/I

%method 1 for C:
slope=8.3e5;
%V=t*I/C % initial slope: no current to resistor.
Ci=I/slope

%method 2 for C:
RC=0.965e-6;
Ci=RC/R

```

Solution A.4.17

```

DiffSlopeInput=4
SlopeOutput=32

```

```
A=SlopeOutput/DiffSlopeInput;  
Acm=0.1/0.2;  
CMRR=A/Acm;
```

```
A=20*log10(A);  
Acm=20*log10(Acm);  
CMRR=20*log10(CMRR);
```


Appendix B

Proofs and deductions

B.1 Linear Circuit Analysis Basics

B.1.1 Stability

Proof of equation (5.35)

repeating (5.35):

$$\frac{1}{|A * B|} > 1 \Leftrightarrow \frac{2(r^2 - 1)}{(r^2 + 1)^2} > \text{Im}\left(\frac{j\omega}{\omega_{p1,2}}\right)^2 \quad (\text{B.1})$$

Defining:

$$r := \frac{\left| \text{Re}\left(\frac{j\omega}{\omega_{p1,2}}\right) \right|}{\text{Im}\left(\frac{j\omega}{\omega_{p1,2}}\right)} \quad (\text{B.2})$$

$$a := \left| \text{Re}\left(\frac{j\omega}{\omega_{p1,2}}\right) \right| > \text{Im}\left(\frac{j\omega}{\omega_{p1,2}}\right) =: b \quad (\text{B.3})$$

$$\begin{aligned} A &:= 1 + \frac{j\omega}{\omega_{p1}} \\ B &:= 1 + \frac{j\omega}{\omega_{p2}} \end{aligned} \quad (\text{B.4})$$

And some observations:

$$\begin{aligned} Q &> \frac{1}{\sqrt{2}} \\ &\Leftrightarrow \\ &\angle \frac{1}{\omega_{p1}} < -45^\circ \\ &\angle \frac{1}{\omega_{p2}} > 45^\circ \\ &\angle \frac{j\omega}{\omega_{p2}} - \angle \frac{j\omega}{\omega_{p1}} > 90^\circ \end{aligned} \quad (\text{B.5})$$

and most importantly:

$$Q > \frac{1}{\sqrt{2}} \Leftrightarrow r > 1 \quad (\text{B.6})$$

Then can be shown that:

$$\begin{aligned}
|A * B|^2 &= ((1+a)^2 + b^2) * ((1-a)^2 + b^2) \\
&= (1+a)^2 * (1-a)^2 + (1+a)^2 * b^2 + b^2 * (1-a)^2 + b^2 * b^2 \\
&= (1+2a+a^2) * (1-2a+a^2) + (1+2a+a^2) * b^2 + b^2 * (1-2a+a^2) + b^4 \\
&= (1+2a+a^2) * (1-2a+a^2) + (1+a^2) * b^2 + b^2 * (1+a^2) + b^4 \\
&= (1+2a+a^2) * (1-2a+a^2) + 2b^2 + 2a^2b^2 + b^4 \\
&= (1 * (1-2a+a^2) + 2a * (1-2a+a^2) + a^2 * (1-2a+a^2)) + 2b^2 + 2a^2b^2 + b^4 \\
&= 1 - 2a + a^2 + 2a * 1 - 2a * 2a + 2a * a^2 + a^2 * 1 - a^2 * 2a + a^2 * a^2 + 2b^2 + 2a^2b^2 + b^4 \\
&= 1 - 2a + a^2 + 2a - 4a^2 + 2a^3 + a^2 - 2a^3 + a^4 + 2b^2 + 2a^2b^2 + b^4 \\
&= 1 + 2a^2 - 4a^2 + a^4 + 2b^2 + 2a^2b^2 + b^4
\end{aligned} \tag{B.7}$$

in what range can this be smaller than 1?

$$\begin{aligned}
1 &> 1 + 2a^2 - 4a^2 + a^4 + 2b^2 + 2a^2b^2 + b^4 \\
0 &> 2a^2 - 4a^2 + a^4 + 2b^2 + 2a^2b^2 + b^4 \\
4a^2 &> 2a^2 + a^4 + 2b^2 + 2a^2b^2 + b^4
\end{aligned} \tag{B.8}$$

This can be solved for a specific ratio of $\frac{a}{b} = r$?

$$\begin{aligned}
0 &> 2(br)^2 + (br)^4 + 2b^2 + 2(br)^2b^2 + b^4 - 4(br)^2 \\
0 &> 2b^2r^2 + b^4r^4 + 2b^2 + 2b^2r^2b^2 + b^4 - 4b^2r^2 \\
0 &> (r^4 + 2r^2 + 1)b^4 + (2r^2 + 2 - 4r^2)b^2 \\
0 &> (r^4 + 2r^2 + 1)b^4 + (2 - 2r^2)b^2 \\
0 &> (r^4 + 2r^2 + 1)b^2 + (2 - 2r^2) \\
(2r^2 - 2) &> (r^4 + 2r^2 + 1)b^2 \\
\frac{(2r^2 - 2)}{(r^4 + 2r^2 + 1)} &> b^2 \\
\frac{2(r^2 - 1)}{(r^2 + 1)^2} &> b^2
\end{aligned} \tag{B.9}$$

$$\frac{1}{|A * B|} > 1 \Leftrightarrow \frac{2(r^2 - 1)}{(r^2 + 1)^2} > \text{Im}\left(\frac{j\omega}{\omega_{p1,2}}\right)^2 \tag{B.10}$$

q.e.d.

B.2 Analog Circuit Basics

B.2.1 Single transistor amplifier LF/DC analysis

Common Gate (CG) gain stage G_M

For deducing G_M , ignore v_{sig} and R_S : starting point as ideal source is v_i . Also ignore R_L but connect v_o directly to Gnd and consider i_o the current flowing

from Gnd into the node output node v_o (do not misunderstand: v_o is zero, so we use 'node v_o ' here to indicate the output node, NOT any voltage).

Kirchhoff for the output node:

$$v_i(g_m + \frac{1}{r_{ds}}) + i_o = 0 \quad (\text{B.11})$$

Solving for G_M defined as $G_M := \frac{i_o}{v_i}$:

$$G_M := \frac{i_o}{v_i} = -(g_m + \frac{1}{r_{ds}}) \quad (\text{B.12})$$

Common Gate (CG) gain stage R_I

For deducing R_I , ignore v_{sig} and R_S and have i_i flowing directly into node v_i . Note that $v_{gs}g_m$ becomes $-v_i g_m$, or if you turn the arrow of that current source around you have $v_i g_m$ flowing from node v_i to node v_o .

Kirchhoff current equation for node v_i :

$$v_i(\frac{1}{r_{ds}} + g_m) = i_i + v_o \frac{1}{r_{ds}} \quad (\text{B.13})$$

Kirchhoff current equation for node v_o :

$$v_o(\frac{1}{r_{ds}} + \frac{1}{R_L}) = v_i(\frac{1}{r_{ds}} + g_m) \quad (\text{B.14})$$

Solve (B.14) for v_o :

$$v_o = v_i \frac{\frac{1}{r_{ds}} + g_m}{\frac{1}{r_{ds}} + \frac{1}{R_L}} \quad (\text{B.15})$$

Substitute into (B.13):

$$v_i(\frac{1}{r_{ds}} + g_m) = i_i + v_i \frac{\frac{1}{r_{ds}} + g_m}{\frac{1}{r_{ds}} + \frac{1}{R_L}} \frac{1}{r_{ds}} \quad (\text{B.16})$$

Multiply with $\frac{1}{r_{ds}} + \frac{1}{R_L}$:

$$v_i(\frac{1}{r_{ds}} + g_m)(\frac{1}{r_{ds}} + \frac{1}{R_L}) = i_i(\frac{1}{r_{ds}} + \frac{1}{R_L}) + v_i(\frac{1}{r_{ds}} + g_m)\frac{1}{r_{ds}} \quad (\text{B.17})$$

Subtract $v_i(\frac{1}{r_{ds}} + g_m)\frac{1}{r_{ds}}$:

$$v_i(\frac{1}{r_{ds}} + g_m)\frac{1}{R_L} = i_i(\frac{1}{r_{ds}} + \frac{1}{R_L}) \quad (\text{B.18})$$

Multiply with $r_{ds}R_L$:

$$v_i(1 + g_m r_{ds}) = i_i(R_L + r_{ds}) \quad (\text{B.19})$$

Solve for R_I defined as $R_I := \frac{v_i}{i_i}$:

$$R_I := \frac{v_i}{i_i} = \frac{R_L + r_{ds}}{1 + g_m r_{ds}} \quad (\text{B.20})$$

Common Gate (CG) gain stage R_O

For deducing R_O , ignore R_L . An ideal source for i_o flowing into node v_o is the starting point. v_{sig} is tied to Gnd.

Kirchhoff for the output node v_o :

$$v_o \frac{1}{r_{ds}} = i_o + v_i \left(\frac{1}{r_{ds}} + g_m \right) \quad (\text{B.21})$$

Kirchhoff current equation for node v_i :

$$v_i \left(\frac{1}{r_{ds}} + \frac{1}{R_S} + g_m \right) = v_o \frac{1}{r_{ds}} \quad (\text{B.22})$$

Solve B.22 for v_i :

$$v_i = v_o \frac{\frac{1}{r_{ds}}}{\frac{1}{r_{ds}} + \frac{1}{R_S} + g_m} \quad (\text{B.23})$$

Substitute into B.21:

$$v_o \frac{1}{r_{ds}} = i_o + v_o \frac{\frac{1}{r_{ds}}}{\frac{1}{r_{ds}} + \frac{1}{R_S} + g_m} \left(\frac{1}{r_{ds}} + g_m \right) \quad (\text{B.24})$$

Multiply with $\frac{1}{r_{ds}} + \frac{1}{R_S} + g_m$

$$v_o \left(\frac{1}{r_{ds}} \right) \left(\frac{1}{r_{ds}} + \frac{1}{R_S} + g_m \right) = i_o \left(\frac{1}{r_{ds}} + \frac{1}{R_S} + g_m \right) + v_o \frac{1}{r_{ds}} \left(\frac{1}{r_{ds}} + g_m \right) \quad (\text{B.25})$$

Subtract $v_o \frac{1}{r_{ds}} \left(\frac{1}{r_{ds}} + g_m \right)$

$$v_o \left(\frac{1}{r_{ds}} \right) \left(\frac{1}{R_S} \right) = i_o \left(\frac{1}{r_{ds}} + \frac{1}{R_S} + g_m \right) \quad (\text{B.26})$$

Multiply with $R_S r_{ds}$

$$v_o = i_o (R_S + r_{ds} + g_m R_S r_{ds}) \quad (\text{B.27})$$

Solve for R_O defined as $R_O := \frac{v_o}{i_o}$:

$$\frac{v_o}{i_o} = R_S + r_{ds} + g_m R_S r_{ds} = r_{ds} + R_S (g_m r_{ds} + 1) \quad (\text{B.28})$$

Appendix C

Matlab code

The course at the University of Oslo where this book has been conceived uses Matlab for the electronic exam. Instead of symbolic expressions, the students are asked to submit Matlab functions that can be tested automatically. Furthermore, instead of numeric results the students are asked to submit matlab scripts that assign the numeric results to a given variable. The scripts can contain the entire deduction of the result instead of just a single number. Thus, should the final result be wrong, the examiner can still give points for correct steps in the deduction process.

This appendix will thus give some examples on how to translate some of the expressions given in this script into Matlab functions. Note that also the exercises (appendix A)) contain examples of both Matlab scripts and Matlab functions.

C.1 The full EKV model

Note that nothing on this level of sophistication will be expected at the exam. This function allows the sweep of any of the three voltage variables, the gate, source, or drain. It's a version where these voltages are referenced to the bulk voltage, i.e. equation (2.7). Note the operators preceded by a '.' perform *element-wise* operations on vectors or matrices of equal dimensions.

The following text needs to be in file EKV.m:

```
function [Id]=EKV(W,L,uCox,UT,Vt,n,Vg,Vs,Vd,lambda,sty)
%function [Id]=EKV(W,L,uCox,UT,Vt,n,Vg,Vs,Vd,lambda,sty)
%Plots the nFET EKV model behaviour. Either Vg, Vs, Vd can be vectors.
% 'sty' is a string defining the line style of the plot (see 'help plot')

beta=uCox*W/L;
Is=2*n*beta*UT^2;

Id=( Is*log(1+exp((Vg-Vt-n*Vs)/(2*n*UT))).^2 ... %forward current
- Is*log(1+exp((Vg-Vt-n*Vd)/(2*n*UT))).^2 ) ... %reverse current
.* (1+lambda*((Vd-Vs) ));%-(Vg-Vt-n*Vs)); %channel length modulation

if (max(size(Vg))>1) %Vg is a vector
plot(Vg,Id,sty);
ylabel('I_D','fontname','times','fontsize',14);
xlabel('V_{G}','fontname','times','fontsize',14);
end;
if (max(size(Vs))>1) %Vs is a vector
plot(Vs,Id,sty);
ylabel('I_D','fontname','times','fontsize',14);
xlabel('V_{S}','fontname','times','fontsize',14);
end;
if (max(size(Vd))>1) %Vd is a vector
plot(Vd,Id,sty);
ylabel('I_D','fontname','times','fontsize',14);
xlabel('V_{D}','fontname','times','fontsize',14);
end;
```

```
% The following command did make a pdf file but the command 'print'
% has since changed and 'convert' is a custom system command at UiO.
% So check 'help print' for a hint on how to do this similarly but correctly.
%
% print -deps2c '../fig/BSIMvsEKV.eps'
% system('convert ../fig/BSIMvsEKV.eps ../fig/BSIMvsEKV.pdf');
end
```