

// Впереди - внезапно лекция на очень низовом уровне, погружаемся обратно в транзисторы, отсечки 0/1, энергопотребление и тд. Хз будет ли так постоянно но крепись

Логические уровни, уровни напряжений, согласование уровней

Двоичная логика

Почему все ЭВМ нынче на цифре? Так тупо проще
Логический уровень - одно из конечных состояний, в которых может находиться цифровой сигнал
Они обычно задаются разностью потенциалов сигнал - земля, пускай и бывают другие стандарты (по типу дифференциальных передач, токовых петель и др.)

Логический уровень	Активный высокий (active-high)	Активный низкий (active-low)
Лог. высокий	1	0
Лог. низкий	0	1
^ как пример - на AVR LED включаются по фронту 0. Как второй пример - нажатие на клавиатуре засчитывается по 0		

Уровни напряжения

Логическим уровням соответствуют уровни напряжения. Простое приближение:

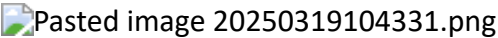
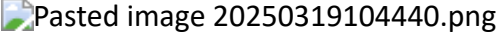
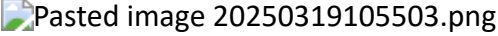
- Меньше порога - 0
- Больше порога - 1

Появляется проблема - близко к порогу может быть много переключений (ихих дребезг схемотехника 5 семестр), то есть по факту всё сложнее

Дальше:

- TTL
- CMOS
- LVC MOS - Low Voltage CMOS

TTL

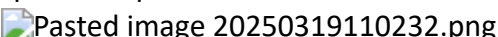
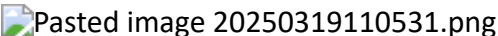

TTL - Transistor-Transistor Logic - одна из первых технологий цифровой логики. Основана на биполярных транзисторах
 ^ TTL-инвертор И-НЕ
 ^ То же самое, эквивалентная схема
Если на выходной транзистор Q3 открывает, то на выходе 0. Если он закрыт, то неопределённое состояние
 ^ Схема с выходом totem-pole
Последняя схема в отличие от двух схем выше:

- Имеет 2 инвертора, соответственно ясен хер сложнее
- На выходе значение 1 формируется принудительно, соответственно на выходе всегда либо 0 либо 1, неопределённых состояний не бывает
Энергопотребление:
- Статическое:
 - TTL потребляет ток даже в статическом состоянии когда входы не меняются. Почему? Потому что биполярные транзисторы требует энергии даже в простое
Стандартный TTL-вентиль потребляет 1-2 мА на каждый вход
- Динамическое:
 - При переключении TTL потребляет больше тока из-за переходных процессов транзисторов, что особенно заметно на высоких частотах

Быстродействие:

- Задержка распространения - около 10-30нс из-за биполярных транзисторов
- Скорость переключения - TTL быстрее ранних CMOS (по типу 4000 серий), но уступает современным (по типу 74НС серии)

CMOS

CMOS (Complementary Metal-Oxide-Semiconductor) - основана на использовании комплементарных пар MOSFET-транзисторов
 ^ Инвертор (NOT-вентиль). По аналогии с totem-pole TTL, это в CMOS называется push-pull
 ^ NAND Gate так для общего развития
"Если Вам хочется всё по ГОСТу - откройте ГОСТ"
© Хохлов 2025
 ^ push-pull vs открытый сток

Преимущество CMOS - в статическом режиме почти не жрёт энергию, ибо один из транзивторов всегда закрыт

Параметр	CMOS	TTL
Статическое потребление	около 0 (только утечка)	1-2 мА на вентиль
Динамическое потребление	Зависит от частоты $P = C * V^2 * f$	Зависит от тока переключения
Быстродействие	Высокое (в современных сериях)	Среднее
Уровни напряжения	Широкий диапазон (1.8-5 В)	0-0.8 В (0), 2-5 В (1)
Помехоустойчивость	Высокая	Низкая
Потребление:		

- CMOS 74HC00 (лог. элемент И-НЕ) - 2.5 мкА в статике, 10 мА при 10 МГц
- TTL 7400 - 10 мА в статике, 22 мА при 10 МГц

CMOS логика почти не потребляет в статике, но при переключении появляются динамические потери из-за заряда/разряда ёмкости затвора (C_{gate})

$$P_{dynamic} = C_{gate} * V^2 * f$$

Как CMOS умудряется работать на высоких частотах?

- Уменьшение ёмкости затвора:
 - Современные технологии (5-7 нм техпроцесс) минимизируют ёмкость затвора за счёт:
 - Уменьшения размеров транзисторов
 - Материалов с высокой подвижностью носителей (кремний-германий например)
 - Пример - C_{gate} техпроцесса 5нм ниже чем C_{gate} техпроцесса 28нм на 30%-50%
- Оптимизация схемотехники
 - Синхронный дизайн - тактовые сигналы синхронизируют переключение, минимизируя задержки
 - Конвейеризация
 - Дифференциальные схемы - LVDS, Current-Mode Logic для уменьшения влияния шумов
- Снижение напряжения питания
 - Применение низковольтных технологий (LVC MOS, Low-Voltage CMOS), что снижает V, что снижает $P_{dynamic}$
- Предусилители и буферы
 - Буферные каскады ускоряют переключение, компенсируя задержки

Примеры:

- Процессоры
 - Тактовые частоты до 5 ГГц (Intel Core i9, AMD Ryzen)
- Телекоммуникации
 - Чипы 5G (28-39 ГГц), SerDes (up to 112 Gb/s)
- Память
 - DDR5 - до 6.4 МГц, GDDR6 - до 20 МГц

LVC MOS

... - модификация стандартной CMOS логики со низким уровнем сигнала и соответственно с более низким потреблением/нагревом/вольтажом

Каким макарон низкий уровень сигнала?

- Снижение напряжения питания - уровни лог. состояний пропорциональны напряжению питания
- Использование MOSFET с малым напряжением отсечки - транзисторы проектируются для эффективной работы на низких напряжениях

Плюсы:

- Энергоэффективность - см. формулу где-то выше: мощность потребления пропорциональна квадрату напряжения. Снижение вольтажа снижает энергопотребление в 2-4 раза по сравнению с 3.3 В
- Меньший нагрев - логично, меньше частоты/напряжение - меньше тепла. Упрощает тепловые расчёты
- Совместимость с современными технологиями - современные процы, память, IoT устройства все используют низковольтные технологии
- Высокая плотность элементов - меньшие размеры транзисторов и межсоединений -> на чипе умещается больше элементов

Минусы:

- Чувствительном к шумам - между логическими уровнями меньше пробел, что означает худшую помехозащищённость
- Требовательность к стабильности питания - нужны точные источники напряжения/фильтрации
- Ограниченная нагрузочная способность - низкий выходной ток требует буферов управления мощными нагрузками

На следующей лекции - LVDS