Полные системы

- Множество B булевых функций называется полной системой, если формулой с множеством операций B можно задать любую булеву функцию
  - ⋆ от любого числа переменных, не меньшего 1
    пример: формула $x \wedge \bar{x}$ задает унарную функцию $f(x)$, равную 0
    пример: формула $x \vee y$ задает не только функцию $f(x, y)$ с таблицей значений

x, y	f
00	0
01	1
10	1
11	1

, но и функцию $g(x, y, z)$ с таблицей значений

x, y, z	g
000	0
001	0
010	1
011	1
100	1
101	1
110	1
111	1

★ Если все функции полной системы B можно задать формулами над множеством функций B', то B' — полная система

⋆ множество $\{\wedge, \vee, \bar{}\}$ является полной системой
- например, по следствию из теоремы об СКНФ (предыдущий фрагмент)

! Какие еще полные системы существуют?
- любое надмножество множества $\{\wedge, \vee, \bar{}\}$
  - включая множества всех булевых функций и всех бинарных булевых функций
- множества $\{\wedge, \bar{}\}$ и $\{\vee, \bar{}\}$
  - выразить $\vee$ ($\wedge$) через две оставшиеся функции по формулам де Моргана
  - сослаться на замечание ★

Стрелка Пирса и штрих Шефера

Напомним, что $x \downarrow y = \overline{x \vee y}$, $x \,'\, y = \overline{x \wedge y}$

Теорема

Множества $\{\downarrow\}$ и $\{\,'\,\}$ являются полными системами.

Доказательство:

- выразим отрицание и дизъюнкцию через стрелку Пирса:
- ⋆ $\bar{x} = x \downarrow x$
- ⋆ $x \vee y = \overline{x \downarrow y} = (x \downarrow y) \downarrow (x \downarrow y)$
- поскольку $\{\vee, \bar{}\}$ — полная система, $\{\downarrow\}$ — тоже полная
- аналогично, отрицание и конъюнкция выражаются через штрих Шефера □

- Верна и обратная теорема:
  - ★ если f — бинарная б.ф. и $\{f\}$ — полная система, то $f \in \{\downarrow, \,'\,\}$
  - обратная теорема следует из теоремы Поста о полноте (докажем потом)
- Еще одну полную систему рассмотрим в следующем фрагменте

Многочлены над $\mathbb{F}_2$

- Поле $\mathbb{F}_2$ — это множество $\{0,1\}$ с операциями $+$ (по $\bmod 2$) и $\cdot$ $(= \wedge)$
  - таблицы операций в привычном виде:

$+$	0	1
0	0	1
1	1	0

$\cdot$	0	1
0	0	0
1	0	1

$\star$ Многочлены от k переменных над $\mathbb{F}_2$ — это k-местные булевы функции

$\star$ Множество функций $\{+, \cdot, 1\}$ — полная система
  - из нее получается $\{\wedge, \bar{}\}$, так как $x \wedge y = xy$, $\bar{x} = x + 1$

$\Rightarrow$ любую функцию можно записать формулой над $\{+, \cdot, 1\}$

- Заметим, что
  - пользуясь коммутативностью и дистрибутивностью, можно раскрывать скобки и приводить подобные слагаемые
  - в $\mathbb{F}_2$ выполняются тождества $xx = x$ и $x + x = 0$

$\Rightarrow$ Любая формула над $\{+, \cdot, 1\}$ эквивалентна многочлену, в котором
  - каждый одночлен — это произведение переменных, в котором все переменные различны, либо свободный член (1 или 0)
  - все одночлены различны

- Описанный канонический вид многочлена называется полиномом Жегалкина

$\star$ Для однозначности записи договоримся, что
  - алфавит переменных Σ — упорядочен
  - в каждом одночлене переменные записываются по возрастанию
  - одночлены записываются по возрастанию в радиксном порядке на Σ^*

Теорема

Любая булева функция задается полиномом Жегалкина, и притом единственным.

Доказательство:

- существование полинома следует из полноты системы $\{+, \cdot, 1\}$ и эквивалентности любой формулы над этой системой полиному Жегалкина (предыдущий слайд)

Единственность: зафиксируем алфавит переменных $\Sigma = \{x_1, \ldots, x_k\}$

- над этим алфавитом существует 2^{2^k} различных булевых функций
  - ⋆ таблица значений б.ф. задается битовым вектором длины 2^k
- одночлены над Σ биективно отображаются на подмножества Σ

$\Rightarrow$ существует 2^k различных одночленов над Σ

- полиномы Жегалкина над Σ биективно отображаются на множества одночленов
  - полиному 0 сопоставим пустое множество одночленов

$\Rightarrow$ существует 2^{2^k} различных полиномов Жегалкина над Σ

★ функция, которая каждому полиному Жегалкина над Σ ставит в соответствие задаваемую им б.ф. — сюръекция

- так как каждая функция задается полиномом Жегалкина

★ сюръекция между двумя конечными множествами одной мощности является инъекцией

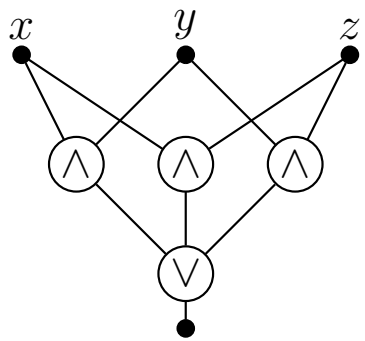
- каждая функция задается единственным полиномом Жегалкина □

⋆ Полином Жегалкина — это нормальная форма («алгебраическая»)

Булевы схемы

- Булева схема (circuit) — альтернативный способ задания булевых функций
  - абстрагирует конструкцию электрической схемы из элементов (вентили, gates)
- Булеву функцию $f(x_1, \ldots, x_k)$ вычисляет черный ящик
  - у ящика k входящих проводов $(x_1, \ldots, x_k)$ и один выходящий (f)
  - ток, идущий по проводу, означает 1, отсутствие тока — 0
  - если токи во входящих проводах соответствуют вектору $(b_1, \ldots, b_k)$, то ток в выходном проводе кодирует $f(b_1, \ldots, b_k)$
- Внутри черного ящика находятся элементы, соединенные проводами в определенном порядке между собой, со входами и выходами
  - каждый элемент — это черный ящик, реализующий одну из функций полной системы B (базы)
  - в реальных электрических и электронных схемах элементы — это физические устройства, такие как реле в дверном звонке или диоды в электронных часах
  - мы рассматриваем идеальные элементы, абстрагируясь от физических сущностей

Пример: функцию большинства от трех переменных можно задать формулой $T_2(x, y, z) = (x \wedge y) \vee (x \wedge z) \vee (y \wedge z)$, которая представляется схемой



$T_2(x, y, z)$

Булевы вектор-функции. Сложение столбиком

- Булева вектор-функция — это произвольная функция $\vec{f} : \{0,1\}^n \to \{0,1\}^m$
  - сложение двух n-битных чисел — это функция $ADD_n : \{0,1\}^{2n} \to \{0,1\}^{n+1}$
  - ⋆ вектор-функции намного удобнее задавать схемами, чем формулами
    - у схемы для вектор-функции m выходов вместо одного
- ⋆ Научимся вычислять функцию ADD_n
  - пусть $a = a_{n-1} \cdots a_0$, $b = b_{n-1} \cdots b_0$ — числа в двоичной записи
    - ведущие нули разрешены
  - $ADD_n(a_{n-1}, \ldots, a_0, b_{n-1}, \ldots, b_0) = (s_n, \ldots, s_0)$, где $s = s_n \cdots s_0 = a + b$
- ⋆ Вычисление столбиком:
  - пусть $c_n, \ldots, c_0$ — вспомогательные булевы переменные, c_i = перенос в разряд i
  - ⋆ $c_0 = 0$, $s_0 = a_0 + b_0$ (сложение по mod 2!)
  - ⋆ $c_i = T_2(a_{i-1}, b_{i-1}, c_{i-1})$ для $i = 1, \ldots, n$ (почему?)
  - ⋆ $s_i = a_i + b_i + c_i$ для $i = 1, \ldots, n-1$; $s_n = c_n$
- Приведенный алгоритм выполняет $\Theta(n)$ операций
- Как и любой другой алгоритм сложения n-битных чисел

  . . . но есть нюанс . . .
- булева схема — это ациклический орграф
- электрический ток способен течь по проводам параллельно, давая возможность параллельного вычисления значений в разных узлах схемы (вершинах графа)
  - ★ время вычисления функции булевой схемой определяется глубиной схемы — максимальной длиной пути от входа до выхода
- Глубина схемы, построенной по алгоритму сложения столбиком, равна $\Theta(n)$
  - ⋆ никакой выгоды от распараллеливания мы не получаем

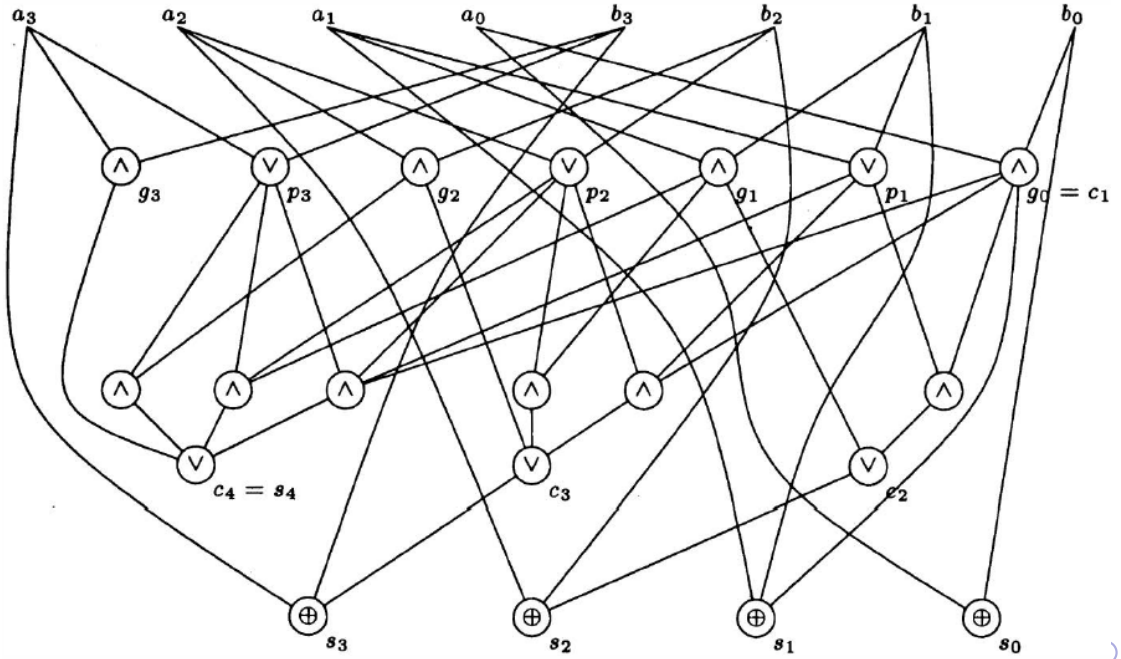
Параллельная схема для сложения

- Проблема сложения столбиком — в последовательном вычислении переносов
  - если все переносы известны, все биты s_i вычисляются параллельно за один шаг
- Рассмотрим эволюцию переносов:
  - разряд i порождает перенос, если $c_i = 0$ и $c_{i+1} = 1$
    - ⋆ тогда $a_i = b_i = 1$, т.е. $a_i \wedge b_i = 1$
  - разряд i сохраняет перенос, если $c_i = 1$ и $c_{i+1} = 1$
    - ⋆ тогда $a_i \vee b_i = 1$

  $\Rightarrow$ $c_i = 1 \Leftrightarrow$ найдется разряд $j < i$ такой, что
  - j порождает перенос, а каждый разряд k, $j < k < i$, сохраняет его
- Положим $p_i = a_i \vee b_i$, $g_i = a_i \wedge b_i \Rightarrow c_i = \bigvee_{j=0}^{i-1} \left(g_j \wedge \bigwedge_{k=j+1}^{i-1} p_k \right)$

★ ADD_n вычисляется за три шага:
шаг 1 — все p_i и g_i; шаг 2 — все c_i; шаг 3 — все s_i

Пример: схема сложения для 4-битных чисел ($n = 4$)



Замкнутые классы

- Пусть B — некоторое множество булевых функций
- ⋆ $\langle B \rangle$ — множество функций, которые можно записать формулами над B
- ⋆ $\langle \cdot \rangle$ — оператор замыкания:
  - $B \subseteq \langle B \rangle$ (экстенсивность)
  - $A \subseteq B \Rightarrow \langle A \rangle \subseteq \langle B \rangle$ (монотонность)
  - $\langle \langle B \rangle \rangle = \langle B \rangle$ (идемпотентность)
- B называется замкнутым классом (булевых функций), если $B = \langle B \rangle$
- ⋆ B — полная система $\Leftrightarrow \langle B \rangle$ содержит все булевы функции

- Б.ф. f сохраняет 0, если $f(\vec{0}) = 0$, и сохраняет 1, если $f(\vec{1}) = 1$
  - множество всех б.ф., сохраняющих 0 (сохраняющих 1) обозначается $\mathbf{T}_0$ ($\mathbf{T}_1$)
    примеры: $0, \vee, \wedge, + \in \mathbf{T}_0$; $1, \bar{}, \sim, \downarrow \notin \mathbf{T}_0$; $1, \vee, \wedge, \sim \in \mathbf{T}_1$; $0, \bar{}, +, ' \notin \mathbf{T}_1$

Лемма

$\mathbf{T}_0$ и $\mathbf{T}_1$ — замкнутые классы.

Доказательство: рассмотрим формулу над $\mathbf{T}_0$, построим по ней схему
- если любому элементу схемы подать 0 на все входы, то на выходе у него будет 0
- подадим 0 на все входы схемы

$\Rightarrow$ на выходе схемы будет 0
$\Rightarrow$ функция, задаваемая схемой, принадлежит $\mathbf{T}_0$

- для $\mathbf{T}_1$ доказательство аналогично

Линейные функции

- Функция $f(x_1, \ldots, x_k)$ линейна, если ее полином Жегалкина — линейный
  - т.е. $f(x_1, \ldots, x_k) = a_0 + a_1 x_1 + a_2 x_2 + \cdots a_k x_k$ для некоторых $a_0, \ldots, a_k \in \{0, 1\}$
  - ⋆ f обладает свойствами самой обычной линейной функции из курса алгебры
  - множество всех линейных б.ф. обозначается **L**

    примеры: $0, \bar{}, +, \sim \in \mathbf{L}$; $\quad \wedge, \vee, \to, \downarrow \notin \mathbf{L}$

Лемма

L — замкнутый класс.

Доказательство: рассмотрим формулу над **L**, построим по ней схему

- каждый элемент схемы вычисляет линейную функцию своих входов
- линейная функция от линейных функций переменных является линейной функцией этих переменных

$\Rightarrow$ вся схема вычисляет линейную функцию $\square$

Самодвойственные функции

- Функция $f(x_1, \ldots, x_k)$ самодвойственна, если $f(\bar{x}_1, \ldots, \bar{x}_k) = \overline{f(x_1, \ldots, x_k)}$
  - на противоположных наборах аргументов f принимает разные значения
  - множество всех самодвойственных б.ф. обозначается **S**
    примеры: $^-, x + y + z, T_2(x, y, z) \in \mathbf{S}$; $0, \vee, \to, \downarrow \notin \mathbf{S}$

Лемма

S — замкнутый класс.

Доказательство: рассмотрим формулу над **S**, построим по ней схему

- подадим на входы произвольный битовый вектор
- ⋆ на выходе каждого элемента схемы будет некоторый бит
- поменяем биты на всех входах
- ⋆ докажем, что бит на выходе каждого элемента поменялся
  индукцией по максимальной длине n пути от входа до элемента
- база индукции: $n = 1$
- входы элемента являются входами схемы, элемент задает функцию из **S**

⇒ выходной бит изменился, так как поменялись все входы

- шаг индукции:
- входами элемента являются либо входы схемы (поменялись по условию), либо выходы элементов с меньшей длиной пути (поменялись по предположению индукции)

⇒ выход элемента, задающего самодвойственную функцию, поменялся

⇒ в частности, поменялся выходной бит всей схемы

⇒ так как рассуждение верно для любого вектора на входе схемы, схема вычисляет самодвойственную функцию

Монотонные функции

- Введем на битовых векторах равной длины покомпонентный порядок:
  - $(x_1, \ldots, x_k) \leqslant (y_1, \ldots, y_k) \Leftrightarrow x_1 \leqslant y_1, \ldots, x_k \leqslant y_k$
  - диаграмма Хассе ЧУМа $(\{0,1\}^k, \leqslant)$ — k-мерный куб
- Функция $f(\vec{x})$ монотонна, если $f(\vec{x}) \leqslant f(\vec{y})$ для любых $\vec{x} \leqslant \vec{y}$
  - если значения каких-то аргументов f увеличить (подняться вверх по кубу), то значение f не уменьшится
  - множество всех монотонных б.ф. обозначается $\mathbf{M}$
    примеры: $0, \vee, \wedge, T_i \in \mathbf{M}$; $\quad +, \bar{}, \to, ' \notin \mathbf{M}$

Лемма

$\mathbf{M}$ — замкнутый класс.

Доказательство: рассмотрим формулу над $\mathbf{M}$, построим по ней схему

- подадим на входы произвольный битовый вектор, не равный $\vec{1}$
- ⋆ на выходе каждого элемента схемы будет некоторый бит
- поменяем биты на некоторых входах с 0 на 1
- ⋆ докажем, что ни у какого элемента выходной бит не поменялся с 1 на 0 индукцией по максимальной длине n пути от входа до элемента
- ! восстановите детали по аналогии с предыдущей леммой

⇒ выходной бит всей схемы не уменьшился

⇒ так как рассуждение верно для любого вектора на входе схемы, схема вычисляет монотонную функцию

□