# Символическая проверка моделей

## Занятие №9

*Любая схема представляется в виде системы уравнений, составленных из символов, соответствующим различным реле и переключателям схемы. <…> этот аппарат в точности аналогичен исчислению высказываний символической логики.*

К. Шеннон.
Символический анализ релейных и переключательных схем

Александр Сергеевич Камкин

kamkin@ispras.ru

# Классическая проверка моделей

- **Явное представление модели**
  - Исследование пространства состояний (поиск в графе)
    - Достижимость состояния с заданным свойством (DFS, BFS)
    - Достижимость допускающего цикла (Tarjan, Nested DFS)
- **Комбинаторный взрыв числа состояний**
  - Оптимизация графа в памяти (сжатие данных)
  - Учет зависимостей между действиями (partial order reduction)
  - Параллельный обход графа состояний

# Символическая проверка моделей

- **Символическое представление модели**
  - Кодировка состояний: $code\colon\ S \to \mathbb{B}^n$
  - Характеристические функции множеств и отношений
    - $\chi_X\colon \mathbb{B}^n \to \mathbb{B} \qquad x \in X \Leftrightarrow \chi_X\big(code(x)\big) = 1$
    - $\chi_R\colon \mathbb{B}^n \times \mathbb{B}^n \to \mathbb{B} \quad (x, x') \in R \Leftrightarrow \chi_R\big(code(x), code(x')\big) = 1$
- **Символические алгоритмы**
  - Представление характеристических функций (КНФ, BDD)
  - Манипуляции с символическими представлениями

# Операции над множествами (состояний)

| Описание | Операция над множествами | Операция над функциями |
|---|---|---|
| ***Константы*** | | |
| **Полное множество** | $S$ | $\chi_S \equiv 1$ |
| **Пустое множество** | $\emptyset$ | $\chi_\emptyset \equiv 0$ |
| ***Унарные операции*** | | |
| **Дополнение множества** | $\overline{X}$ | $\chi_{\bar{X}} \equiv \neg\chi_X$ |
| ***Бинарные операции*** | | |
| **Пересечение множеств** | $X \cap Y$ | $\chi_{X\cap Y} \equiv \chi_X \wedge \chi_Y$ |
| **Объединение множеств** | $X \cup Y$ | $\chi_{X\cup Y} \equiv \chi_X \vee \chi_Y$ |
| **Разность множеств** | $X \setminus Y$ | $\chi_{X\setminus Y} \equiv \chi_X \wedge \neg\chi_Y$ |

# Операции над отношениями (переходов)

| Описание | Операция над отношениями | Операция над функциями |
|---|---|---|
| ***Образ и домен отношения*** | | |
| **Образ отношения** | $Im(R) = \{y \in S \mid \exists x((x,y) \in R)\}$ | $\chi_{Im(R)}(\vec{x}') = \exists \vec{x}(\chi_R(\vec{x},\vec{x}')) = \bigvee_{\vec{x} \in \mathbb{B}^n} \chi_R(\vec{x},\vec{x}')$ |
| **Домен отношения** | $Dom(R) = \{x \in S \mid \exists y((x,y) \in R)\}$ | $\chi_{Dom(R)}(\vec{x}) = \exists \vec{x}'(\chi_R(\vec{x},\vec{x}')) = \bigvee_{\vec{x}' \in \mathbb{B}^n} \chi_R(\vec{x},\vec{x}')$ |
| ***Ограничения образа и домена*** | | |
| **Ограничение домена** | $X \lhd R = \{(x,y) \in R \mid x \in X\}$ | $\chi_{X \lhd R}(\vec{x},\vec{x}') = \chi_X(\vec{x}) \wedge \chi_R(\vec{x},\vec{x}')$ |
| **Ограничение образа** | $R \rhd Y = \{(x,y) \in R \mid y \in Y\}$ | $\chi_{R \rhd Y}(\vec{x},\vec{x}') = \chi_R(\vec{x},\vec{x}') \wedge \chi_Y(\vec{x}')$ |
| ***Образ (прообраз) множества относительно отношения*** | | |
| **Прямой образ** | $R(X) = Im(X \lhd R)$ | $\chi_{R(X)}(\vec{x}') = \exists \vec{x}(\chi_X(\vec{x}) \wedge \chi_R(\vec{x},\vec{x}'))$ |
| **Обратный образ** | $R^{-1}(Y) = Dom(R \rhd Y)$ | $\chi_{R^{-1}(Y)}(\vec{x}) = \exists \vec{x}'(\chi_R(\vec{x},\vec{x}') \wedge \chi_Y(\vec{x}'))$ |

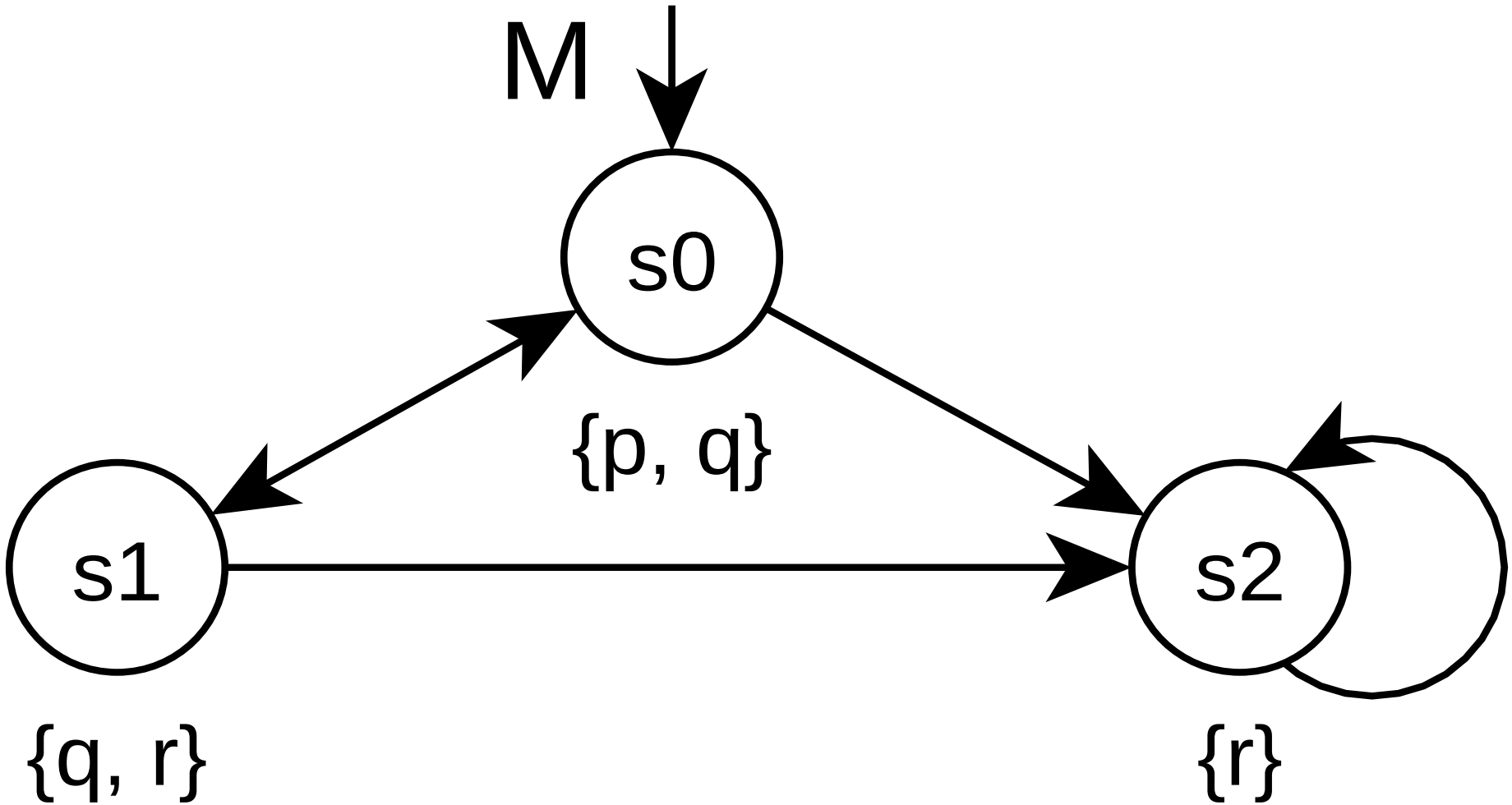
# Представление структуры Крипке

$$\langle code, \chi_0, \chi_R, \{\chi_p\}_{p \in AP} \rangle$$

- $code: S \to \mathbb{B}^{\log |S|}$ – кодировка состояний
- $\chi_0$ – характеристическая функция множества $S_0$
- $\chi_R$ – характеристическая функция отношения $R$
- $\chi_p$ – характеристическая функция множества

$$\{s \in S \mid p \in L(s)\}$$

# Пример: представление структуры Крипке



# Пример: кодировка состояний и $\chi_0$

| Состояние $s$ | Двоичная кодировка $code(s)$ | | Характеристическая функция $\chi_s$ |
|---|---|---|---|
| | $x_1$ | $x_2$ | |
| $s_0$ | 0 | 0 | $\neg x_1 \wedge \neg x_2$ |
| $s_1$ | 0 | 1 | $\neg x_1 \wedge x_2$ |
| $s_2$ | 1 | 0 | $x_1 \wedge \neg x_2$ |

$$\chi_S(x_1, x_2) \equiv (\neg x_1 \wedge \neg x_2) \vee (\neg x_1 \wedge x_2) \vee (x_1 \wedge \neg x_2)$$

$$\chi_0(x_1, x_2) \equiv (\neg x_1 \wedge \neg x_2)$$

# Пример: характеристическая функция $\chi_R$

| Таблица истинности $\chi_R$ | | | | |
|---|---|---|---|---|
| $x_1$ | $x_2$ | $x_1'$ | $x_2'$ | $\chi_R$ |
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 0 |

| Символическое представление $\chi_R$ | |
|---|---|
| Совершенная ДНФ | Упрощенная формула |
| | |
| $\neg x_1 \wedge \neg x_2 \wedge \neg x_1' \wedge x_2'$ | $\neg x_1 \wedge \neg x_2 \wedge (x_1' \oplus x_2')$ |
| $\neg x_1 \wedge \neg x_2 \wedge x_1' \wedge \neg x_2'$ | |
| | |
| $\neg x_1 \wedge x_2 \wedge \neg x_1' \wedge \neg x_2'$ | $\neg x_1 \wedge x_2 \wedge \neg x_2'$ |
| | |
| $\neg x_1 \wedge x_2 \wedge x_1' \wedge \neg x_2'$ | |
| | |
| | |
| | |
| $x_1 \wedge \neg x_2 \wedge x_1' \wedge \neg x_2'$ | $x_1 \wedge \neg x_2 \wedge x_1' \wedge \neg x_2'$ |
| | |
| | |
| | |
| | |
| | |

# Пример: характеристические функции $\{\chi_p\}_{p \in AP}$

| Состояние $s$ | Двоичная кодировка $code(s)$ | | Характеристическая функция $\chi_s$ |
|---|---|---|---|
| | $x_1$ | $x_2$ | |
| $s_0$ | 0 | 0 | $\neg x_1 \wedge \neg x_2$ |
| $s_1$ | 0 | 1 | $\neg x_1 \wedge x_2$ |
| $s_2$ | 1 | 0 | $x_1 \wedge \neg x_2$ |

| Элементарное высказывание $\varphi$ | Множество $[\![\varphi]\!]$ | Характеристическая функция $\chi_\varphi$ |
|---|---|---|
| $p$ | $\{s_0\}$ | $\neg x_1 \wedge \neg x_2$ |
| $q$ | $\{s_0, s_1\}$ | $\neg x_1$ |
| $r$ | $\{s_1, s_2\}$ | $x_1 \oplus x_2$ |

# Символические алгоритмы на графах

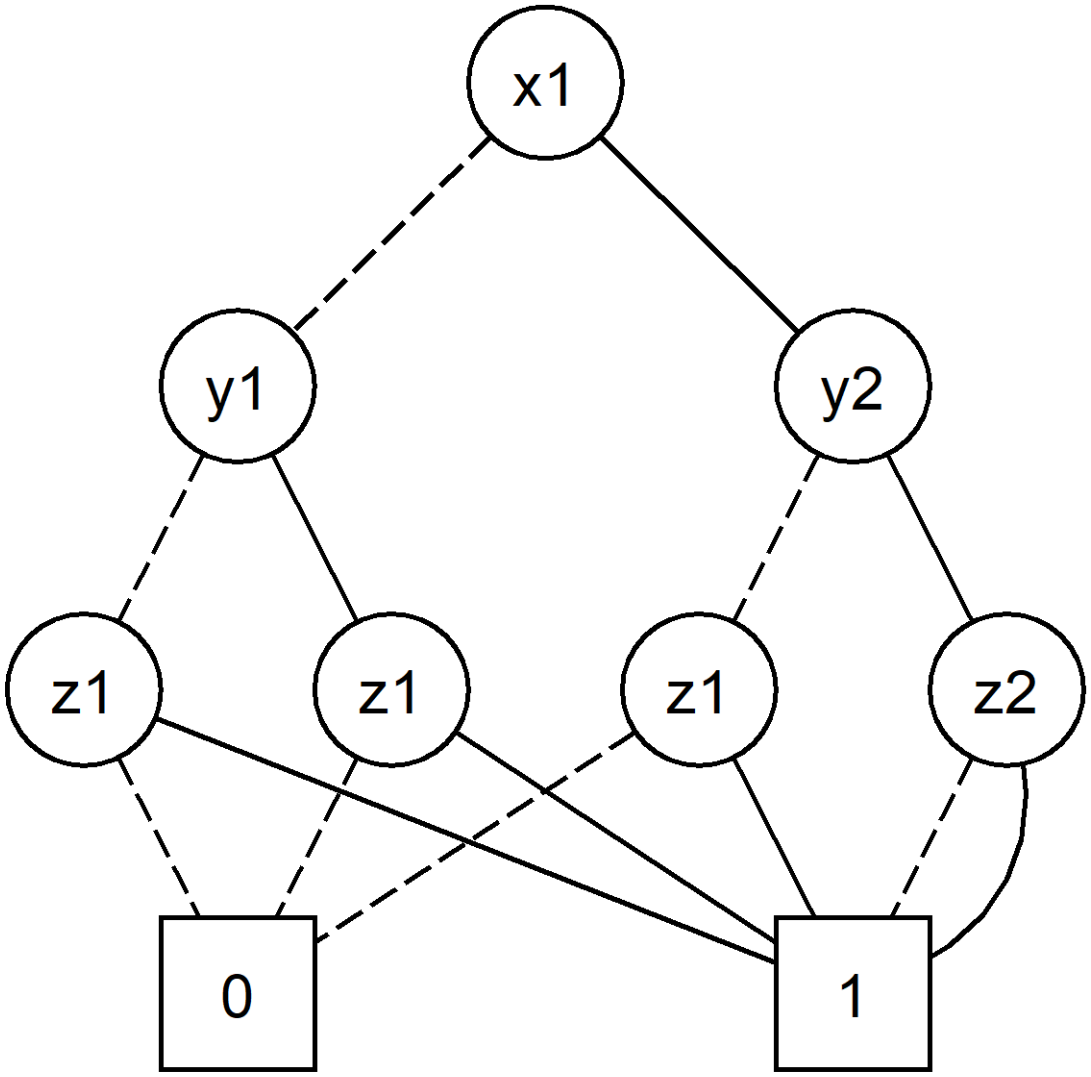
| Обычный алгоритм | Символический алгоритм |
| --- | --- |
| $S_{old} := S_0$; | $\chi_{old} := \chi_0$; |
| $S_{new} := S_{old} \cup R(S_{old})$; | $\chi_{new} := \chi_{old} \vee \chi_{R(old)}[x' := x]$; |
| **while** $S_{new} \neq S_{old}$ **do** | **while** $\chi_{new} \neq \chi_{old}$ **do** |
| $S_{old} := S_{new}$; | $\chi_{old} := \chi_{new}$; |
| $S_{new} := S_{old} \cup R(S_{old})$ | $\chi_{new} := \chi_{old} \vee \chi_{R(old)}[x' := x]$ |
| **end** | **end** |

# Двоичные решающие диаграммы (BDD)

$$\langle N \cup \{\boxed{0}, \boxed{1}\}, n_0, L, low, high \rangle$$

- $N \cup \{\boxed{0}, \boxed{1}\}$ — множество вершин ($N \cap \{\boxed{0}, \boxed{1}\} = \emptyset$)
- $n_0 \in N \cup \{\boxed{0}, \boxed{1}\}$ — корень
- $L: N \to V$ — разметка вершин переменными
- $low: N \to N \cup \{\boxed{0}, \boxed{1}\}$ — $low$-дуги
- $high: N \to N \cup \{\boxed{0}, \boxed{1}\}$ — $high$-дуги

# Пример двоичной решающей диаграммы




$$(\neg x_1 \wedge \neg y_1 \wedge z_1) \vee$$
$$(\neg x_1 \wedge y_1 \wedge z_1) \vee$$
$$(x_1 \wedge \neg y_2 \wedge z_1) \vee$$
$$(x_1 \wedge y_2 \wedge \neg z_2) \vee$$
$$(x_1 \wedge y_2 \wedge z_2)$$

# Семантика двоичных решающих диаграмм

- если $root(F) = \boxed{0}$, то $[\![F]\!] \equiv 0$ — константа 0
- если $root(F) = \boxed{1}$, то $[\![F]\!] \equiv 1$ — константа 1
- иначе $[\![F]\!] \equiv (L(F) \wedge [\![then(F)]\!]) \vee (\neg L(F) \wedge [\![else(F)]\!])$

*Разложение Шеннона*

  - $root(F)$ — корень диаграммы $F$
  - $L(F)$ — переменная, помечающая корень $F$
  - $then(F)$ — поддиаграмма $F$ с корнем $high\big(root(F)\big)$
  - $else(F)$ — поддиаграмма $F$ с корнем $low\big(root(F)\big)$

# Сокращенные упорядоченные BDD (ROBDD)

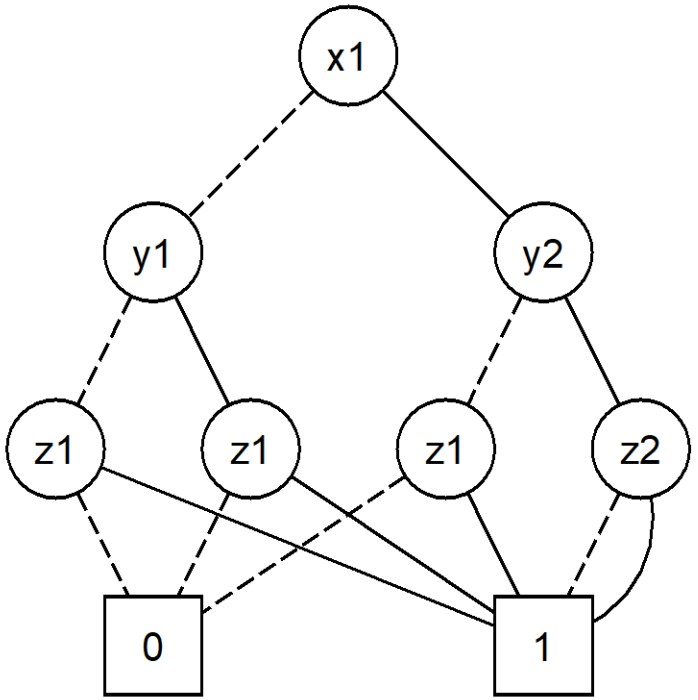
- **Упорядоченная (ordered) BDD**
  - на множестве переменных задан линейный порядок $\prec$
  - $L(n) \prec L(n')$, если $n'$ — потомок $n$
- **Сокращенная (reduced) BDD**
  - отсутствуют *изоморфные подграфы*
  - отсутствуют *избыточные вершины*
    - $low(n) = high(n)$

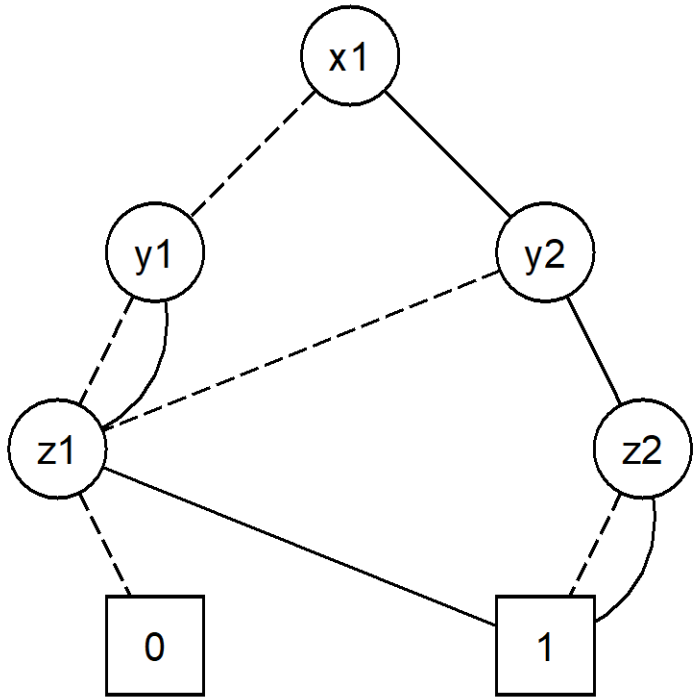
Randal E. Bryant
(1952)

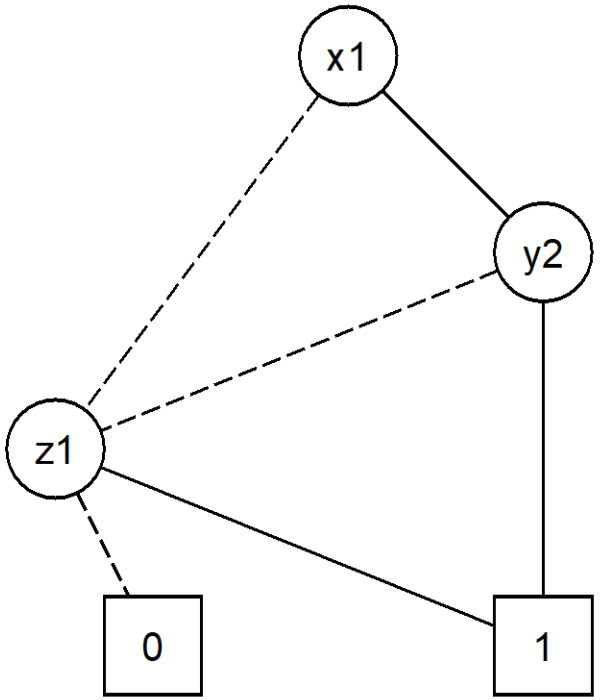
# Построение сокращенной BDD

- если есть два изоморфных подграфа
  - удалить один из подграфов
  - перенаправить ведущие в него дуги в соответствующие вершины оставшегося подграфа
- если есть избыточная вершина
  - удалить вершину
  - перенаправить ведущие в нее дуги в ее последователя

# Пример: построение сокращенной BDD



# Манипуляции с ROBDD

$$Apply(F \circ G) = Reduce\left(Compose\left(x, Apply\left(F_{|x=1} \circ G_{|x=1}\right), Apply\left(F_{|x=0} \circ G_{|x=0}\right)\right)\right)$$

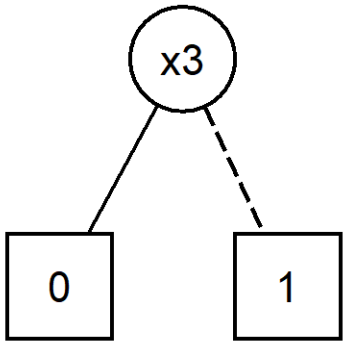
- $Reduce(F)$ — применение правил сокращения к диаграмме $F$
- $Compose(x, T, E)$ — композиция диаграмм $T$ и $E$ по переменной $x$
- $F_{|x=\sigma}$ — диаграмма, полученную из $F$:
  - для каждой вершины $n$, такой что $L(n) = x$, дуги, входящие в $n$, перенаправляются в
    - $low(n)$, если $\sigma = 0$
    - $high(n)$, если $\sigma = 1$

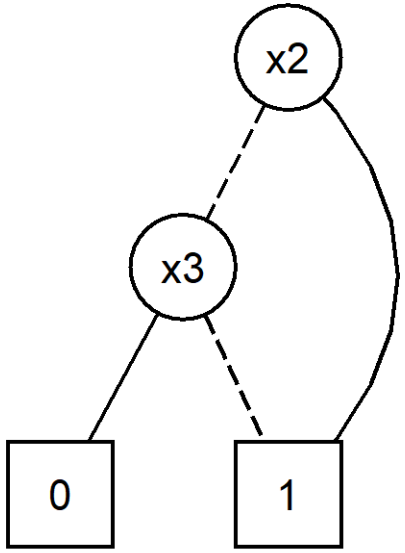
$x$ – минимальная из корневых переменных

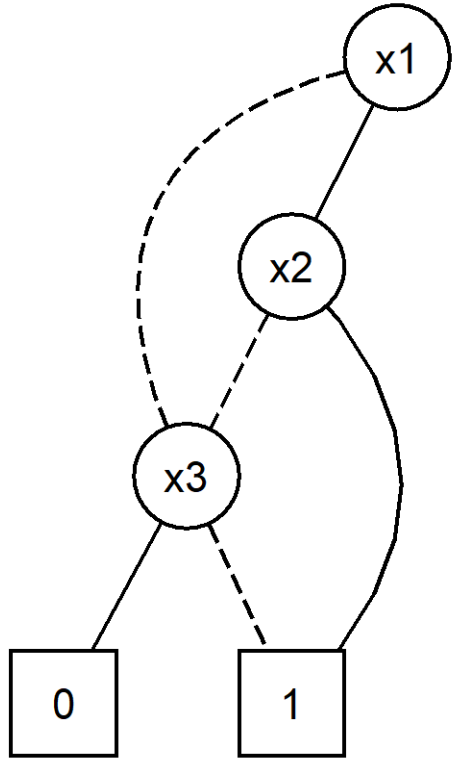
$$F_{|x=\sigma} = \begin{cases} then(F), & \text{если } x = L(F) \text{ и } \sigma = 1 \\ else(F), & \text{если } x = L(F) \text{ и } \sigma = 0 \\ F, & \text{если } x \prec L(F) \end{cases}$$

# Пример: ROBDD $(x_1 \wedge x_2) \vee \neg x_3$

- $Apply\big((x_1 \wedge x_2) \vee \neg x_3\big) = Compose\big(x_1, Apply(x_2 \vee \neg x_3), Apply(\neg x_3)\big)$
- $Apply(x_2 \vee \neg x_3) = Compose\big(x_2, ONE, Apply(\neg x_3)\big)$
- $Apply(\neg x_3) = Compose(x_3, ZERO, ONE)$



# Символическая проверка моделей

- Построение представления $\mathcal{R}(B_M)$
- Построение представления $\mathcal{R}\left(B_{\neg\varphi}\right)$
- Построение представления $\mathcal{R}\left(B_M \otimes B_{\neg\varphi}\right)$
- Проверка языка $\mathcal{L}_{B_M \otimes B_{\neg\varphi}}$ на пустоту

# Символическое представление модели

| | | |
|---|---|---|
| (1) | $\mathcal{R}(l\colon \mathbf{skip}, l')$ | $(pc = l) \wedge (pc' = l') \wedge same(V)$ |
| (2) | $\mathcal{R}(l\colon x \coloneqq t, l')$ | $(pc = l) \wedge (pc' = l') \wedge (x' = t) \wedge same(V \setminus \{x\})$ |
| (3) | $\mathcal{R}(l_1\colon P_1; \ldots; l_n\colon P_n, l_{n+1})$ | $\bigvee_{i=1}^{n} \mathcal{R}\,(l_i\colon P_i, l_{i+1})$ |
| (4) | $\mathcal{R}(l\colon \mathbf{if}\ B\ \mathbf{then}\ l_1\colon P_1\ \mathbf{else}\ l_2\colon P_2\ \mathbf{end}, l')$ | $\big((pc = l) \wedge B \wedge (pc' = l_1) \wedge same(V)\big) \vee \mathcal{R}(l_1\colon P_1, l') \vee$ $\big((pc = l) \wedge \neg B \wedge (pc' = l_2) \wedge same(V)\big) \vee \mathcal{R}(l_2\colon P_2, l')$ |
| (5) | $\mathcal{R}(l\colon \mathbf{while}\ B\ \mathbf{do}\ l_1\colon P_1\ \mathbf{end}, l')$ | $\big((pc = l) \wedge B \wedge (pc' = l_1) \wedge same(V)\big) \vee \mathcal{R}(l_1\colon P_1, l) \vee$ $\big((pc = l) \wedge \neg B \wedge (pc' = l') \wedge same(V)\big)$ |

# Представление параллельной программы

- Синхронная композиция
  - $\mathcal{R}(P) = \Lambda_{i=1}^{n} \mathcal{R}(P_i)$
- Асинхронная композиция
  - $\mathcal{R}(P) = (p \in \{1, \dots, n\}) \wedge \left(\Lambda_{i=1}^{n}\left((p = i) \rightarrow \mathcal{R}(P_i)\right)\right)$

# Поиск достижимых циклов (SCC Hull)

$S_{old} := \emptyset;$

$S_{new} := R^*(S_0);$

**while** $S_{new} \neq S_{old}$ **do**

  $S_{old} := S_{new};$

  $S_{new} := R^*(S_{new} \cap F);$

  **while** $S_{new} \neq (S_{new} \cap R(S_{new}))$ **do**

    $S_{new} := S_{new} \cap R(S_{new})$

  **end**

**end;**

$verdict := (S_{new} = \emptyset)$

# Построение контрпримера ($S_{source} \to S_{target}$)

$\pi := \varnothing;$

$S_{start} := S_{source};$

**while** $(S_{start} \cap S_{target}) = \varnothing$ **do**

  $S_{end} := R^{-1}(S_{target});$

  **while** $(S_{start} \cap S_{end}) = \varnothing$ **do**

    $S_{end} := R^{-1}(S_{end})$

  **end;**

  $s := \text{choose}(S_{start} \cap S_{end});$

  $\pi := \pi \cdot \{s\};$

  $S_{start} := R(s)$

**end**