

# Введение в автоматное программирование

---

**А. А. Шалыто**

Санкт-Петербургский государственный университет  
информационных технологий, механики и оптики

2009 г.

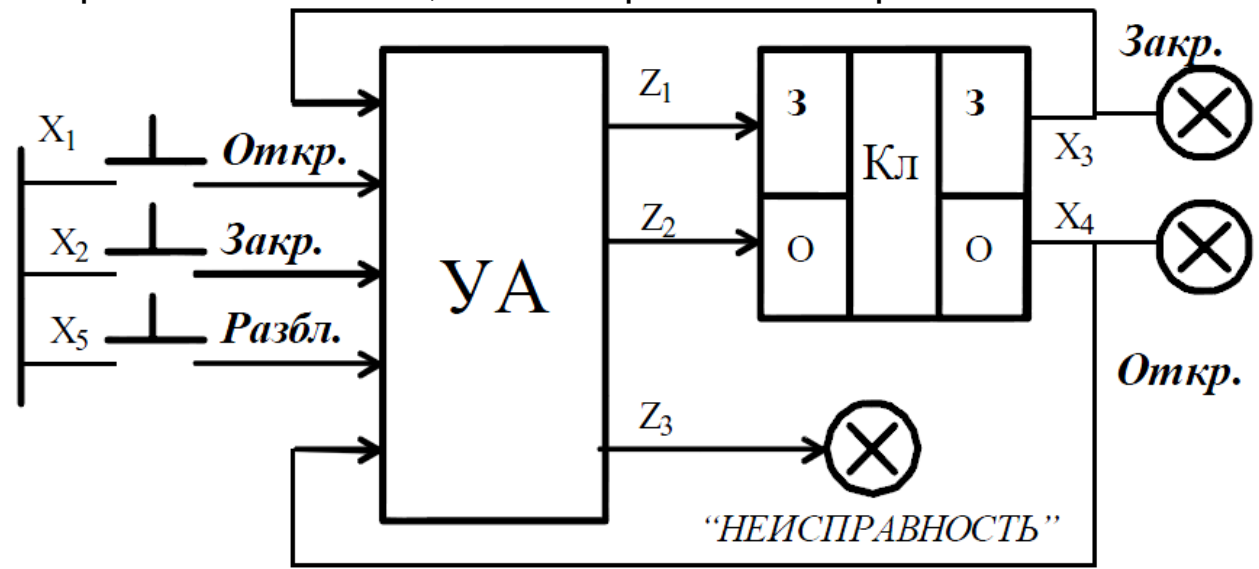
# 1. Словесное описание

Приведем словесное описание алгоритма управления клапаном:

1. При нажатии кнопки «Откр.» клапан начинает открываться.
2. После его открытия срабатывает сигнализатор открытого положения, зажигается лампа «Откр.» и управляющий сигнал с клапана снимается.
3. При нажатии кнопки «Закр.» клапан начинает закрываться.
4. После его закрытия срабатывает сигнализатор закрытого положения, зажигается лампа «Закр.» и управляющий сигнал с клапана снимается.
5. Если в течение трех секунд клапан не откроется или не закроется, то управляющий сигнал с клапана снимается и зажигается лампа контроля «Неисправность».
6. Сброс сигнала контроля осуществляется нажатием кнопки «Разбл.» (Разблокировка).

## 2. Схема связей

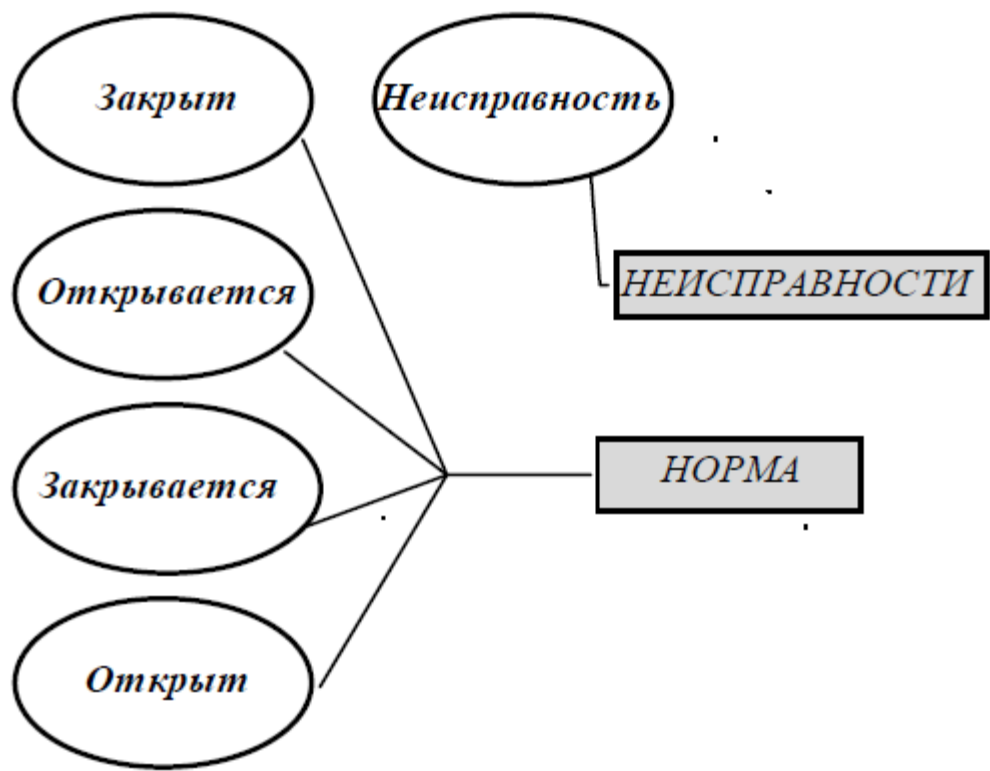
Строится схема связей «источники информации — управляющий автомат — средства представления информации — исполнительные механизмы», являющаяся в общем случае схемой с обратными связями, как это принято в теории автоматического управления.



Каждая информационная связь в схеме помечается переменной. При этом считается, что входные переменные управляющего автомата принадлежат множеству двоичных переменных  $X$ , а его выходные переменные – множеству двоичных переменных  $Z$ .

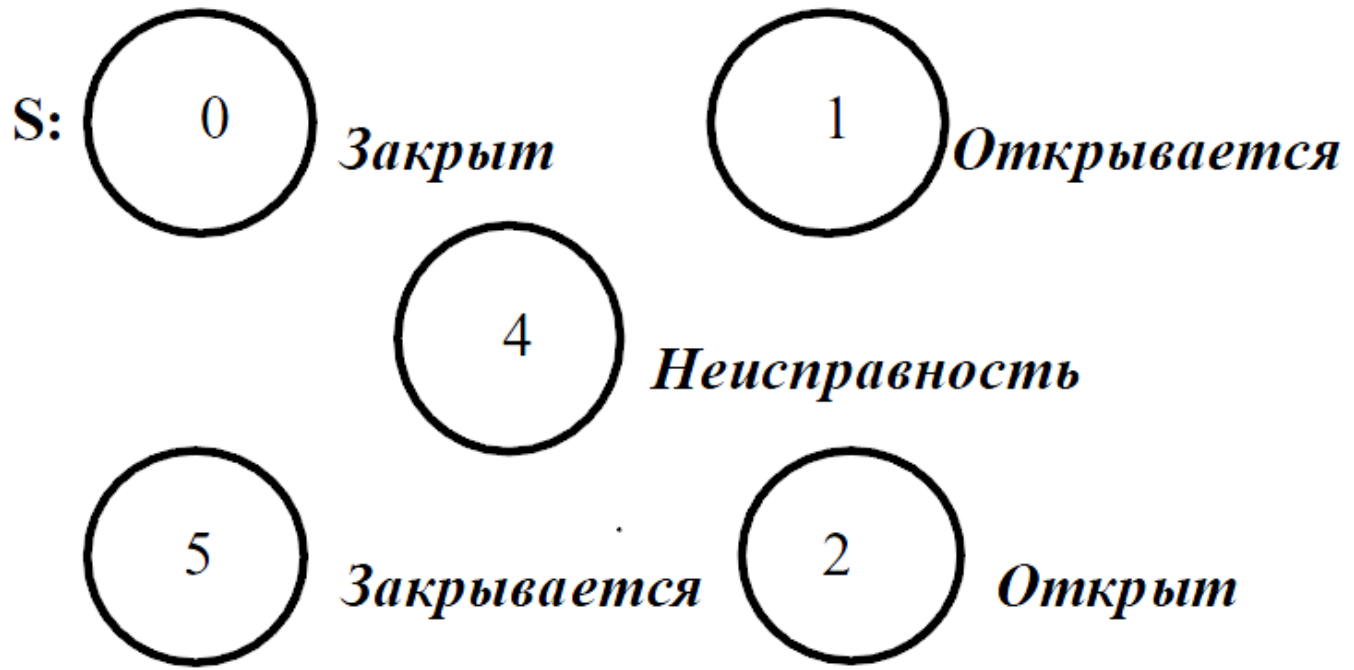
# 3. Состояния объекта

Вводится понятие "состояние" объекта управления. Определяются и перечисляются (классифицируются) его состояния.



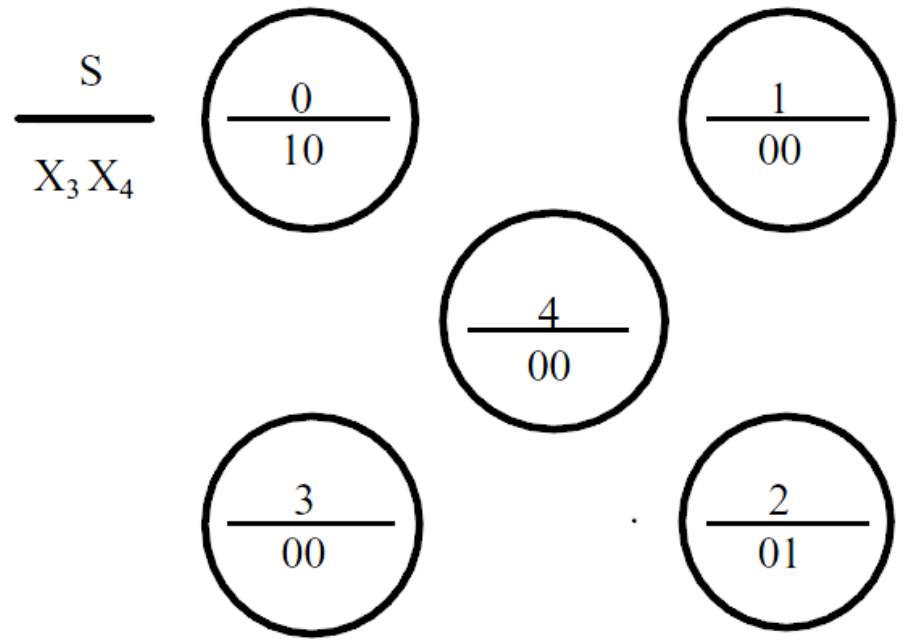
# 4.1. Граф объекта

Каждому состоянию объекта сопоставляется вершина в графе объекта. Вершины располагаются на плоскости и для идентификации нумеруются десятичными числами от 0 до  $S - 1$ , где  $S$  – число выделенных состояний объекта. При этом можно считать, что числа являются значениями одной многозначной переменной  $S$ .



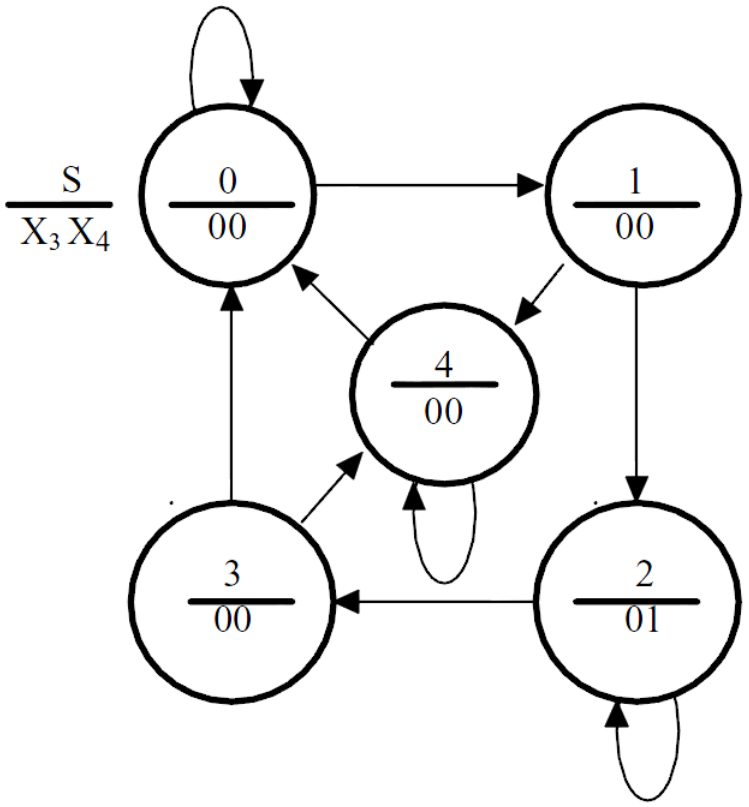
## 4.2. Граф объекта

Каждая вершина графа объекта через дробь с кодирующей ее цифрой помечается кортежем значений двоичных переменных, являющихся подмножеством множества  $X$ , которые формируются сигнализаторами объекта в этом состоянии.



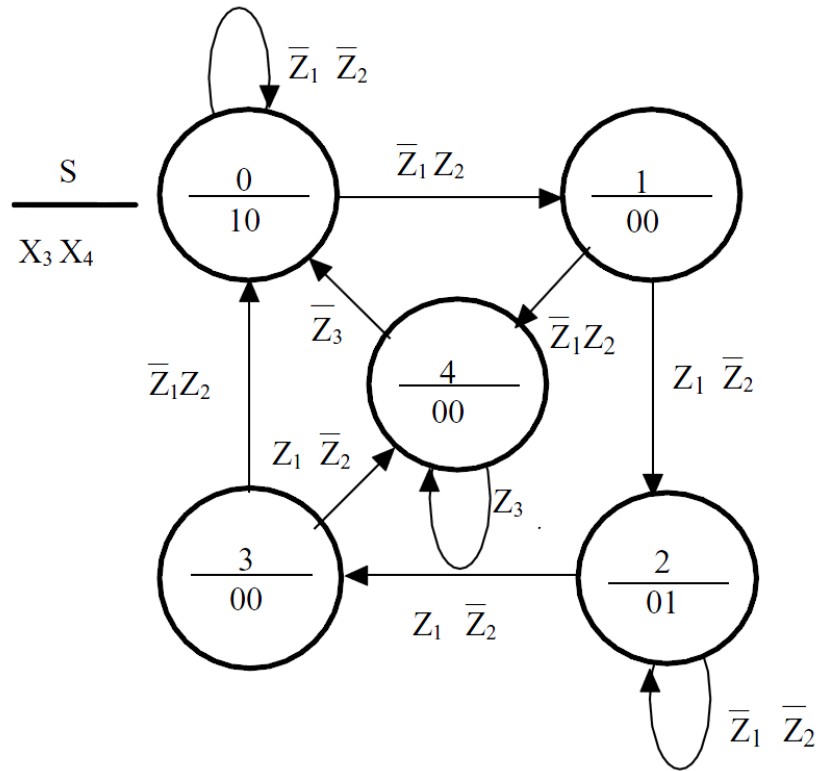
# 4.3. Граф объекта

Определяются все допустимые переходы между состояниями объекта, что отражается введением соответствующих дуг в граф объекта.



# 4.4. Граф объекта

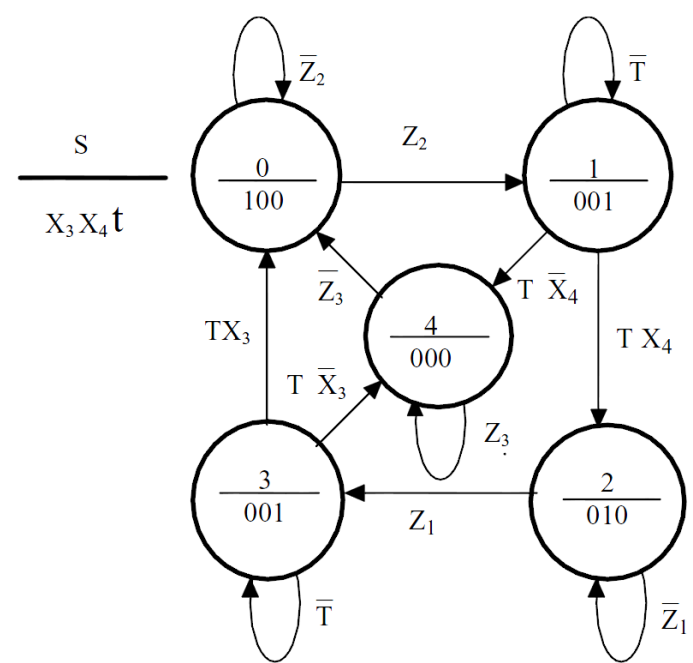
Каждая дуга и петля в графе объекта помечается конъюнкциями переменных или их инверсий из подмножества множества  $Z$ , которые соответствуют значениями переменных, подаваемых на входы исполнительных механизмов объекта и средств представления информации. В этом графе в неустойчивых вершинах исходящие дуги могут быть помечены одинаково.





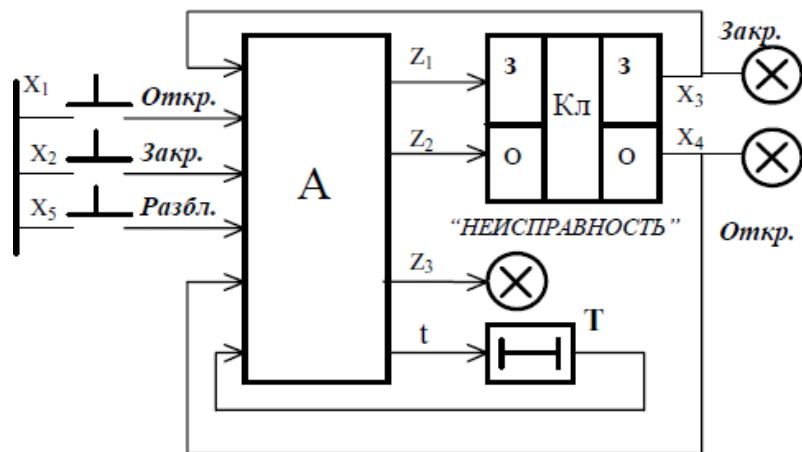
# 5. Переход от объекта к модели объекта

По графу объекта строится модель объекта, которую будем называть графом переходов функционирования модели (граф модели). В графе модели полностью сохраняется «скелет» графа объекта и пометки его вершин. В этом графе все вершины должны быть устойчивыми, и поэтому каждая из них должна иметь инцидентную ей петлю. В каждую вершину к значениям переменных из множества  $X$  добавляются значения двоичных переменных из множества  $t$ , управляющих функциональными элементами задержки, которые моделируют времена пребывания объекта в неустойчивых состояниях. Число этих переменных может быть сведено к одной, если можно сделать предположение о том, что все переходные процессы в объекте имеют одинаковую длительность.



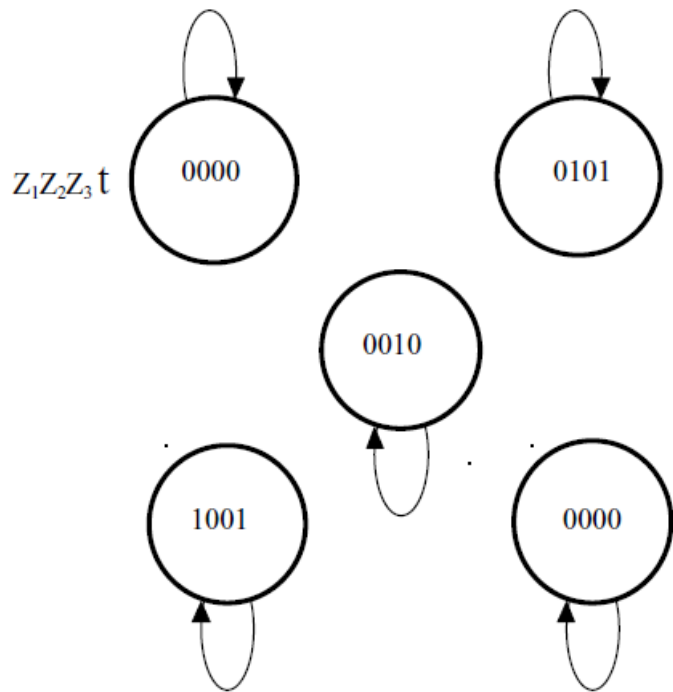
# 6. Время в автоматном программировании

Построение графа переходов автомата начинается с анализа технического задания с целью определения необходимости применения функциональных элементов задержки в алгоритме управления. Если в этих элементах нет необходимости, то строится новая схема связей, отличающаяся от исходной заменой словосочетания "управляющий автомат" на слово "автомат." В случае, когда в задании упоминаются временные задержки, строится новая схема связей, отличающаяся от исходной тем, что в ней управляющий автомат декомпозирован на автомат и функциональный элемент задержки, число которых равно числу различных временных задержек, упоминаемых в задании. При этом для автомата вводится два множества двоичных переменных  $t$  и  $T$ , описанных в предыдущем пункте, первое из которых для автомата является выходным, а второе – входным.



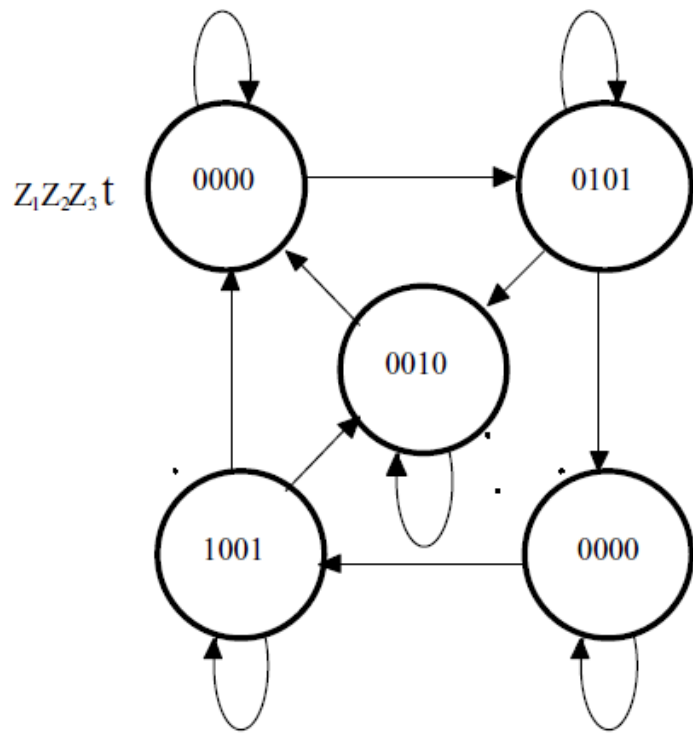
# 7.1. Граф переходов автомата управления

Для каждой вершины графа переходов автомата (граф автомата) определяется кортеж значений всех его выходных переменных, образующих множество  $Z$ , который обеспечивает пребывание объекта в состоянии (устойчивом или неустойчивом), соответствующем рассматриваемому состоянию автомата.



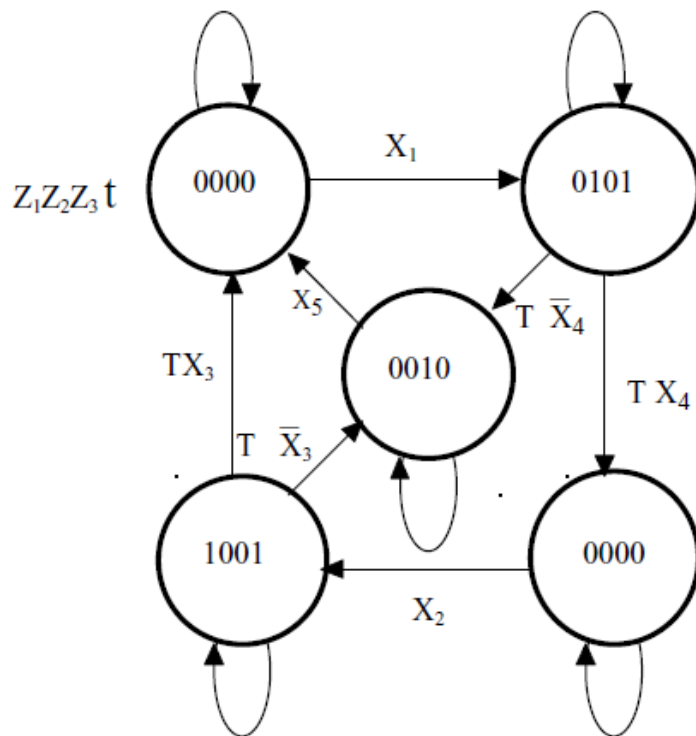
# 7.2. Граф переходов автомата управления

Вершины графа автомата соединяются дугами в соответствии с соединением вершин в графе объекта.



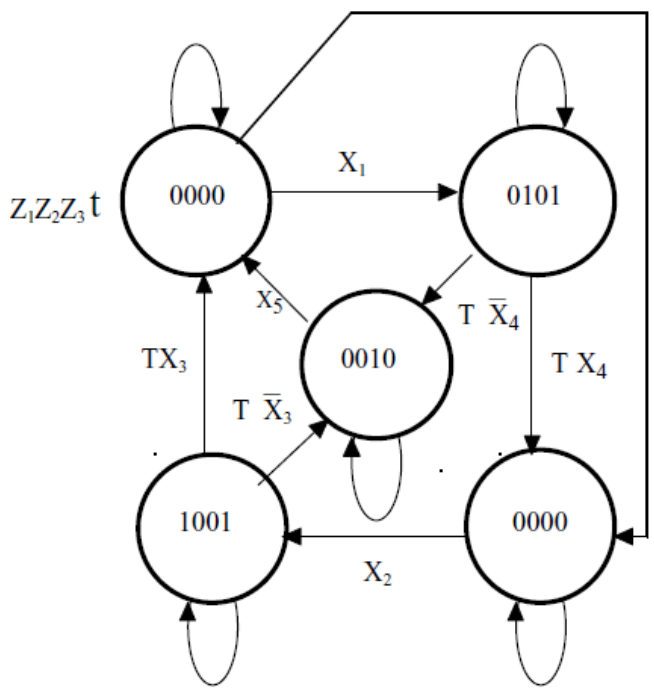
# 7.3. Граф переходов автомата управления

Каждая дуга графа автомата помечается булевой формулой, составленной только из тех переменных множества  $X$  и (или) множества  $T$ , равенство единице которой инициирует соответствующий переход в автомате.



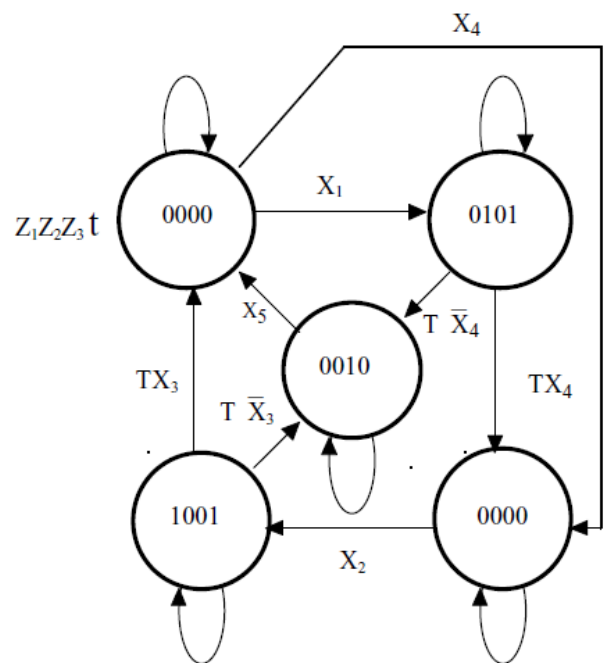
# 7.4. Граф переходов автомата управления

В граф автомата могут быть введены также и дополнительные дуги, отсутствующие в графе объекта. Например, в граф автомата может быть введена дополнительная дуга для устранения возможного несоответствия начальных состояний автомата и объекта.



# 7.5. Граф переходов автомата управления

Если дополнительные дуги вводятся в граф автомата, то каждая из них должна быть помечена булевой формулой. Равенство единице значения этой формулы инициирует переход между вершинами, соединенными этой дугой. Например, несоответствие между состояниями автомата и объекта может быть устранено в результате перехода по введенной дуге при равенстве единице булевой формулы, зависящей от минимально возможного числа переменных, характеризующих исходное состояние объекта.

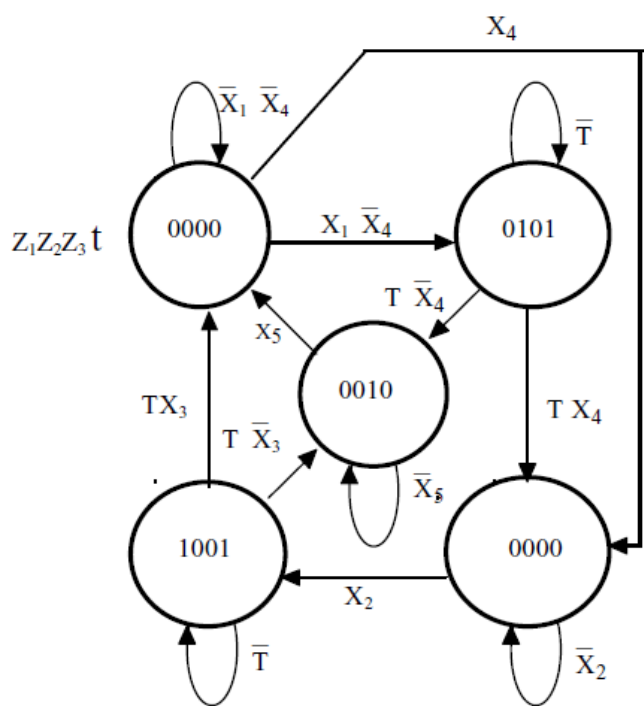






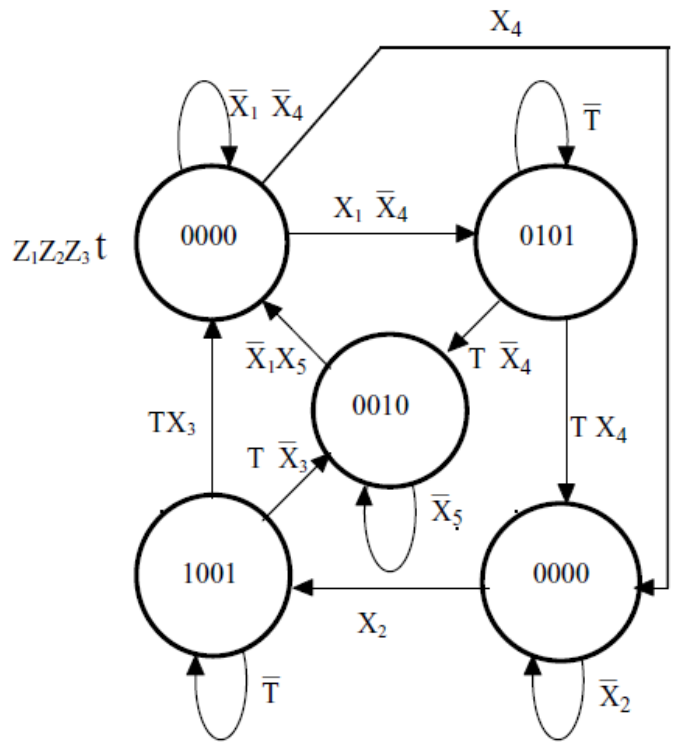
# 9. Полнота

При необходимости осуществляется пометка петель вершин графа автомата за счет обеспечения полноты переходов из каждой вершины.



# 10. Устранение генерирующих контуров

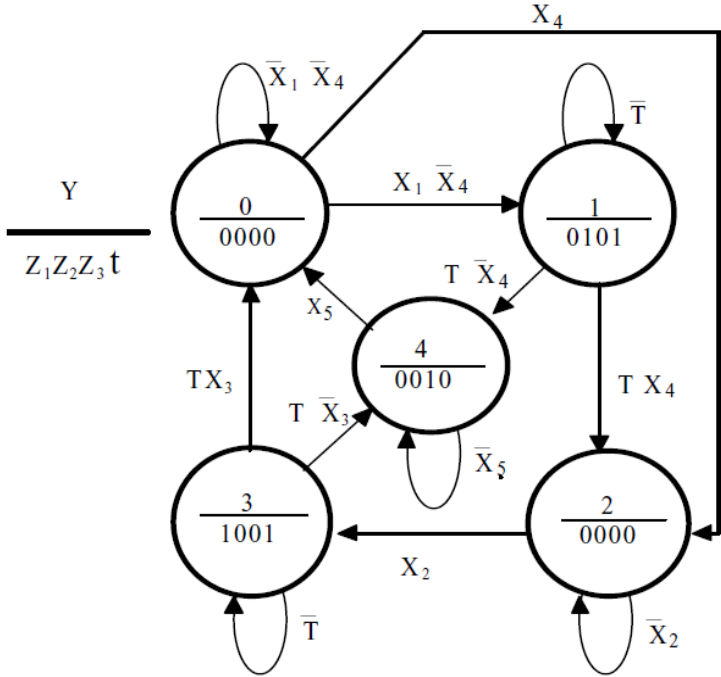
Построенный граф автомата анализируется на наличие генерирующих контуров, отличных от петель, которые устраняются при их обнаружении .



# 11. Кодирование состояний

Если в построенном графе автомата имеются вершины, помеченные одинаково, то для их различения должно использоваться кодирование.

Если в построенном графе автомата все вершины помечены различными кортежами выходных переменных, то эти кортежи могут непосредственно применяться в качестве кодов состояний автомата. Однако, и в этом случае бывает целесообразно закодировать вершины графа переходов, значениями переменной  $Y$ , принимающей значения от 0 до  $S-1$  – сохранить верхнюю пометку вершин графа объекта.



Спасибо за внимание!

<http://is.ifmo.ru>

<http://blog.savethebest.ru>

<http://www.savethebest.ru>

[shalyto@mail.ifmo.ru](mailto:shalyto@mail.ifmo.ru)