Глава IV. Конечные абстрактные автоматы.

4.1. Автоматы с одним входом и выходом

В данной главе будем рассматривать устройства, которые мгновенно преобразовывают поступающие по входным каналам сигналы из некоторого множества. При этом выходные сигналы формируются из заранее определенного множества в зависимости от входных сигналов в текущий момент времени и от состояний, в которые перешло это устройство в предыдущий момент времени. Такие устройства называют ***абстрактными автоматами***. Они являются математическими моделями таких автоматов, как, например, турникеты в метро, игровые автоматы, торговые автоматы (продающие кофе, билеты на транспорт и т.п.).

4.1.1. Начальные понятия теории автоматов

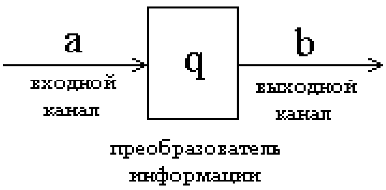
Как следует из вышесказанного, для задания автомата необходимо указать 3 множества:

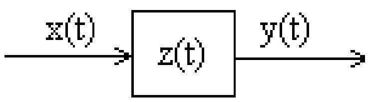
A – множество входных сигналов (входной алфавит),

B – множество выходных сигналов (выходной алфавит),

Q – множество внутренних состояний автомата (алфавит состояний).

Если A, B, Q – конечные множества, то соответствующий автомат называется ***конечным***.





Время работы автомата считается дискретным t = 0, 1, 2, 3, …. . Входные, выходные сигналы и состояния автомата являются функциями от времени:

$a = x(t)$, $b = y(t)$, $q = z(t)$.

Выходные сигналы определяются **функцией выхода**: F: $A\times Q \to B$, т.е. $F(a, q)=b$, где $a\in A$, $b\in B$, $q\in Q$.

За смену состояния автомата отвечает **функция следующего состояния** (функция перехода): G: $A\times Q \to Q$, т.е. $G(a, q) = q'$.

Кроме того, особо выделяют начальное состояние q_0.

Подводя итог вышесказанному, можно дать определение автомата:

Определение 4.1. **Конечным абстрактным автоматом** называется система $\alpha = \langle A, B, Q, F, G, q_0\rangle$.

Способы задания конечного автомата:

1) Табличный

x(t)	z(t-1)	y(t)	z(t)
A	Q	F	G

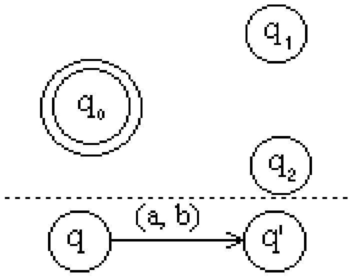
2) Аналитический (формульный) $\begin{cases} y(t) = F(x(t),\ z(t-1)), \\ z(t) = G(x(t),\ z(t-1)), \\ z(0) = q_0. \end{cases}$ - это **канонические уравнения** автомата (другое название - **рекуррентные формулы задания автомата)**, полностью определяющие состояние автомата.

3) Графический – с помощью **диаграммы Мура**.

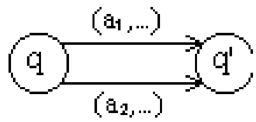
Определение 4.2. **Диаграмма Мура** это ориентированный корневой граф:

(а) множество вершин которого совпадает с множеством Q;

(б) его ребра (a, b), где $a \in$A, $b \in$B; причем для $\forall q \in$ Q и для $\forall a \in$A существует только одно ребро, выходящее из вершины q, в котором первая буква – a.



F(a, q) = b
G(a, q) = q'

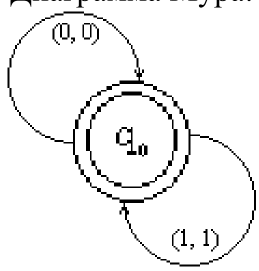


Примеры автоматов:

1) <u>Тривиальный</u> Q = $\{q_0\}$, A=B=E_2={0; 1}.

x(t)	z(t-1)	y(t)	z(t)
0	q_0	0	q_0
1	q_0	1	q_0

Диаграмма Мура:

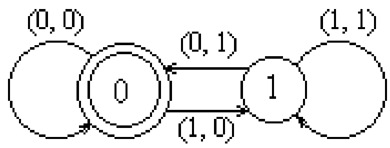


2) <u>Автомат с запаздыванием (автомат-задержка).</u>
A=B=Q={0; 1}

Определение: $\begin{cases} y(t) = z(t-1), \\ z(t) = x(t), \\ z(0) = 0. \end{cases}$ Из определения следует, что $y(t) = x(t-1)$.

x(t)	z(t-1)	y(t)	z(t)
0	0	0	0
0	1	1	0
1	0	0	1
1	1	1	1

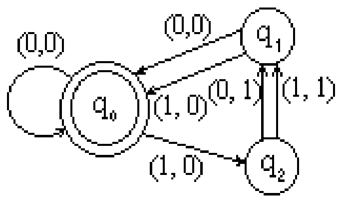
Диаграмма Мура:



3) <u>Автомат с тремя состояниями (покой, возбуждение и промежуточное состояние)</u>:

A=B=E_2, $Q = \left\{ \underbrace{q_0}_{покой}, q_1, \underbrace{q_2}_{возбуждение} \right\}$

x(t)	z(t-1)	y(t)	z(t)
0	q_0	0	q_0
1	q_0	0	q_2
0	q_1	0	q_0
1	q_1	0	q_0
0	q_2	1	q_1
1	q_2	1	q_1



Словарные функции

Пусть A = $\{a_1, \ldots, a_k\}$ – набор букв (алфавит),
$\alpha = \alpha_1\alpha_2\ldots\alpha_n$ – слово ($\alpha_i \in A$), $|\alpha|$ - длина слова (число букв, из которых состоит слово),
λ - пустое слово, $|\lambda| = 0$.

<u>Свойства</u>:

1) Свойство пустого слова: $\lambda\alpha = \alpha\lambda = \alpha$
2) Ассоциативность: $(\alpha\beta)\chi = \alpha(\beta\chi)$
3) Некоммутативность: $\alpha\beta \neq \beta\alpha$

<u>Обозначения</u>:
A^n – множество всех слов длины n.

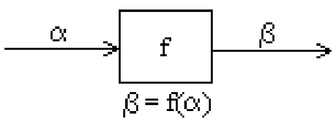
$A^* = \bigcup\limits_{n=0}^{\infty} A^n$ - множество всех слов, которые можно составить из букв алфавита A.

Сверхслово – слово, состоящее из бесконечного числа букв.
A^ω - множество всех сверхслов $\alpha_1\alpha_2\ldots$
Отображение f: $A^* \to B^*$ называется **словарной функцией**.

Примеры:

1) $f(x) = \lambda$
2) $f(x) = x$
3) $f(x_1, \ldots, x_n) = x_n x_{n-1}\ldots x_1$ f(сон) = нос
4) $A=B=\{0, 1\} \Rightarrow f(x_1, \ldots, x_n) = x_1, x_1\vee x_2, x_2\vee x_3, \ldots, x_{n-1}\vee x_n$

Определение 4.3. Словарная функция называется **автоматной**, если существует автомат, реализующий ее.



Покажем, что действительно словарную функцию f: $A^* \to B^*$ можно реализовать автоматом. Введем функцию g: $A^* \to Q$. Пусть $g(\alpha)$ – состояние, в которое перейдет автомат после подачи сигнала α. Тогда $\begin{cases} f(\lambda) = \lambda \\ f(\alpha\beta) = f(\alpha)\,F(\beta, g(\alpha)) \end{cases}$, где F – функция выхода.

$$\begin{cases} g(\lambda) = q_0 \text{ - начальное состояние автомата} \\ g(\alpha\beta) = G(\beta, g(\alpha)) \text{ - функция следующего состояния автомата} \end{cases}$$

Таким образом, соответствие между функцией f и автоматом $\langle A, B, Q, F, G, q_0\rangle$ установлено.

Определение 4.4. Словарная функция f(x) называется **детерминированной**, если выполняются условия:

1) длины выходного и входного слов равны между собой: $|f(\alpha)| = |\alpha|$;
2) для любых слов α, $\beta \in A^*$ слово $f(\alpha)$ – начало слова $f(\alpha\beta)$, то есть $f(\alpha\beta)=f(\alpha)f_\alpha(\beta)$, где $f_\alpha(\beta)$ – остаточная функция для слова α.

Пример:
$f_\alpha(\beta) = \alpha\vee\beta_1, \beta_1\vee\beta_2, \ldots$
$\alpha=0 \Rightarrow f_\alpha(\beta) = \beta_1, \beta_1 \vee \beta_2, \ldots = g(\beta)$
$\alpha=1 \Rightarrow f_\alpha(\beta) = 1, \beta_1\vee\beta_2, \ldots = h(\beta)$
$g(\beta)$ и $h(\beta)$ – разные остаточные функции.

Утверждение: Функция, остаточная к детерминированной, есть детерминированная функция.

Доказательство:

Докажем f_α – детерминированная, если $f(x)$ – детерминированная. Для этого проверим выполнение двух условий из определения детерминированной функции (4.4).

1) $|f_\alpha(\beta)| = |\beta|$?

Так как f детерминированная то:

а) $|f(\alpha,\beta)| = |\alpha\beta| = |\alpha| + |\beta|$ (1)

б) $f(\alpha,\beta) = f(\alpha)f_\alpha(\beta)$ и, следовательно,

$$|f(\alpha,\beta)| = |f(\alpha)| +\cdot|f_\alpha(\beta)| = |\alpha| +\cdot|f_\alpha(\beta)|. \quad (2)$$

Из (1) и (2) следует

$$|f_\alpha(\beta)| = |\beta|$$

2) $f_\alpha(\beta\gamma) = f_\alpha(\beta)\delta$?

Так как f детерминированная, то

$$f(\alpha\beta\gamma) = f(\alpha)f_\alpha(\beta\gamma). \quad (3)$$

С другой стороны,

$$f(\alpha\beta\gamma) = f(\alpha\beta)f_{\alpha\beta}(\gamma) = f(\alpha)f_\alpha(\beta)f_{\alpha\beta}(\gamma). \quad (4)$$

Из (3) и (4) следует:

$$f_\alpha(\beta\gamma) = f_\alpha(\beta)f_{\alpha\beta}(\gamma).$$

$f_{\alpha\beta}(\gamma) = \delta$ - остаточная функция к функции f_α.

Поскольку выполнены оба условия детерминированной функции, то f_α - детерминированная функция.

Вывод: Мы доказали, что если f – детерминированная, то:

1) f_α – детерминированная,
2) $(f_\alpha)_\beta = f_{\alpha\beta}$ - детерминированная,
3) $f(x_1,...,x_n) = f(x_1)f_{x_1}(x_2,...,x_n) = f(x_1)f_{x_1}(x_2)f_{x_1x_2}(x_3,...,x_n) =$
$= f(x_1)f_{x_1}(x_2)...f_{x_1...x_{n-1}}(x_n)$.

Определение 4.5. Число остаточных функций для данной функции f называется ее **весом/** Обозначение **-** $r(f)$.

Определение 4.6. Детерминированная функция называется **ограниченно-детерминированной**, если ее вес конечен.

Теорема (Критерий автоматности функции)

Функция является автоматной тогда и только тогда, когда она ограниченно детерминированная.

Доказательство:

1) Необходимость. Так как функция автоматная, то по определению существует автомат, реализующий ее. Строим диаграмму Мура. Число вершин (состояний автомата) конечно, но оно не может быть меньше веса функции. Значит, вес конечен. Следовательно, функция является ограниченно-детерминированной по определению.
2) Достаточность (без доказательства).

Определение 4.7. Функция f: $A^\omega \rightarrow B^\omega$ называется **детерминированной** тогда и только тогда, когда выполняются условия:

1) Для любого s ≥ 0, s-й символ $y(s)$ слова $\tilde{y}^{\omega} = f\left(\tilde{x}^{\omega}\right)$ - однозначная функция первых s символов $x(1), x(2),\ldots x(s)$ слова $\tilde{x}^{\omega}$.
2) Если начала слов $\tilde{x}_1^{\omega}$ и $\tilde{x}_2^{\omega}$ совпадают, то у слов $\tilde{y}_1^{\omega} = f\left(\tilde{x}_1^{\omega}\right)$ и $\tilde{y}_2^{\omega} = f\left(\tilde{x}_2^{\omega}\right)$ совпадают начала той же длины.

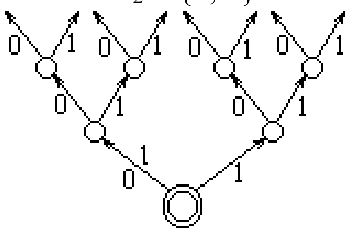
Другими словами, функция $f(x_1, x_2, \ldots, x_n,\ldots)=y_1, y_2, \ldots, y_n,\ldots$- детерминированная, если y_n зависит от $x_1, x_2, \ldots, x_n$, но не зависит от последующих входных символов $x_{n+1}, x_{n+2},\ldots$

Пример: φ(x(1), x(2), …) = x(2)x(3)…, то есть y(t) = x(t+1), t ≥ 1.

Данная функция не является детерминированной, так как y(t) зависит от входного сигнала в следующий момент времени.

4.1.2. Построение диаграммы Мура для ограниченно-детерминированных функций

$A = B = E_2 = \{0, 1\}$



Каждому ребру припишем 0 или 1 по следующему правилу:
0 – левое ребро
1 – правое ребро

Для любого слова α можно найти только одни ориентированный путь, идущий из корня, двигаясь по которому, можно прочитать слово α.

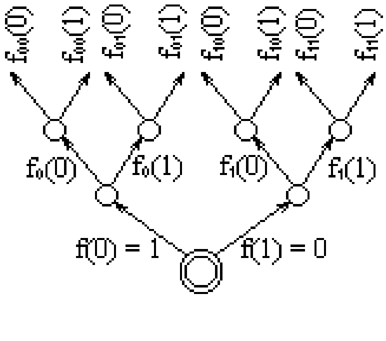
Например, пусть $\pi = e_1e_2\ldots e_n$, $\varphi(\pi) = \varphi(e_1)\ldots\varphi(e_n)$

$$\varphi(e_i) = \begin{cases} 0, e_i - \text{левое ребро}, \\ 1, e_i - \text{правое ребро} \end{cases}$$

Введем другое отображение: $\psi(e_i)$- отображение, определяющее значения функции.

ψ(π) = f(α), где f – некоторая функция, связанная с информационным деревом.

В общем случае информационное дерево имеет вид:



Пример: α = 110, $\underset{f(\alpha)}{f(110)} = \underset{\psi(\pi)}{010}$

1) |f(α)| = 3 = |α|
2) $f(\alpha\beta) = f(\alpha)f_{\alpha}(\beta)$

⇒ f - детерминированная

Утверждение. Функция может быть представлена информационным деревом тогда и только тогда, когда она детерминированная.

$T(v_i) \sim f(\alpha)$ – поддерево, растущее из вершины v_i.

Определение 4.8. Вершины v_1 и v_2 называются эквивалентными, если остаточные функции, соответствующие деревьям $T(v_1)$ и $T(v_2)$ одинаковые.

Если собрать все эквивалентные вершины в класс, то каждый класс будет соответствовать какой-либо остаточной функции. Таких классов будет столько, сколько остаточных функций. Таким образом, функция является ограниченно-детерминированной тогда и только тогда, когда число классов эквивалентных вершин в информационном дереве конечно.

Пример: Является ли словарная функция f: $X^*\to Y^*$, где $X=Y=\{0;\ 1\}$, $f(x_1, x_2, \ldots, x_n, \ldots)=x_1, x_1\oplus x_2, x_1\oplus x_2\oplus x_3, x_2\oplus x_3\oplus x_4, \ldots$ автоматной? Если да, то постройте автомат, реализующий ее.

Решение: Данная функция $f(x_1, x_2, \ldots, x_n, \ldots)$ является детерминированной по определению, так как любое y_i зависит от $x_1, \ldots, x_i$, но не зависит от последующих входных символов. Выясним, является ли она ограниченно-детерминированной. Для этого найдем все ее остаточные функции.

1) $f(0, x_1, ..., x_n, ...) = \underbrace{0}_{f(0)}, \underbrace{0\oplus x_1, 0\oplus x_1 \oplus x_2, x_1\oplus x_2\oplus x_3, ...}_{f_0}$

○ $f_0(x_1, x_2, \ldots) = x_1, x_1\oplus x_2, x_1\oplus x_2\oplus x_3, \ldots = f(x_1, x_2, \ldots)$

$f(1, x_1, ..., x_n, ...) = \underbrace{1}_{f(1)}, \underbrace{1\oplus x_1, 1\oplus x_1 \oplus x_2, x_1\oplus x_2\oplus x_3, ...}_{f_1}$

□ $f_1(x_1, x_2, \ldots) = 1\oplus x_1, 1\oplus x_1 \oplus x_2, x_1\oplus x_2\oplus x_3, \ldots$

2) $f_1(0, x_1, ..., x_n, ...) = \underbrace{1\oplus 0}_{f_1(0)=1}, \underbrace{1\oplus 0\oplus x_1, 0\oplus x_1 \oplus x_2, x_1\oplus x_2\oplus x_3, \ldots}_{f_{10}}$

△ $f_{10}(x_1, x_2, \ldots) = 1\oplus x_1, x_1\oplus x_2, x_1\oplus x_2\oplus x_3, \ldots$

$f_1(1, x_1, ..., x_n, ...) = \underbrace{1\oplus 1}_{f_1(1)=0}, \underbrace{1\oplus 1\oplus x_1, 1\oplus x_1 \oplus x_2, x_1\oplus x_2\oplus x_3, \ldots}_{f_{11}}$

◇ $f_{11}(x_1, x_2, \ldots) = x_1, 1\oplus x_1\oplus x_2, x_1\oplus x_2\oplus x_3, \ldots$

Далее аналогично:

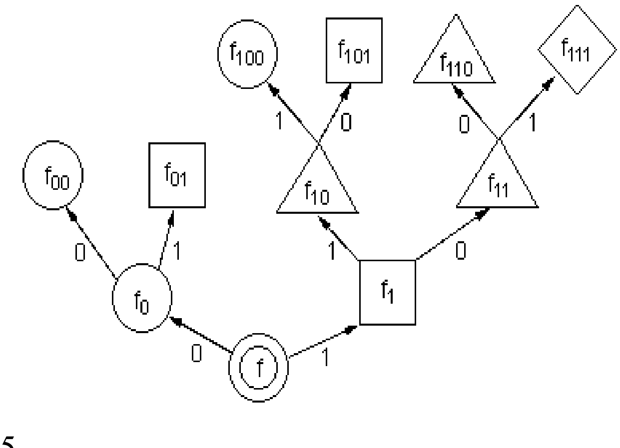
3) ○ $f_{000}(x_1, x_2, \ldots) = x_1, x_1\oplus x_2, x_1\oplus x_2\oplus x_3, \ldots = f(x_1, x_2, \ldots)$

□ $f_{101}(x_1, x_2, \ldots) = 1\oplus x_1, 1\oplus x_1 \oplus x_2, x_1\oplus x_2\oplus x_3, \ldots = f_1(x_1, x_2, \ldots)$

△ $f_{110}(x_1, x_2, \ldots) = 1\oplus x_1, x_1\oplus x_2, x_1\oplus x_2\oplus x_3, \ldots = f_{10}(x_1, x_2, \ldots)$

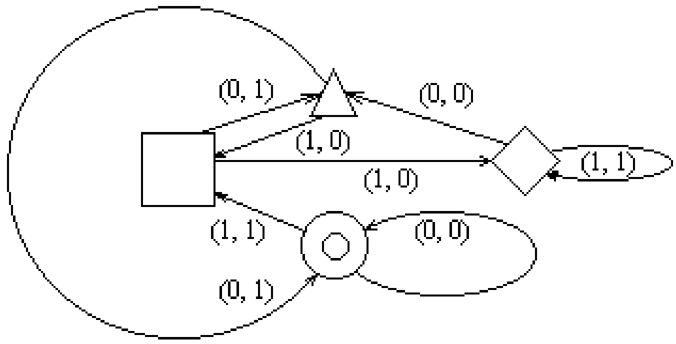
◇ $f_{111}(x_1, x_2, \ldots) = x_1, 1\oplus x_1\oplus x_2, x_1\oplus x_2\oplus x_3, \ldots = f_{11}(x_1, x_2, \ldots)$

Поскольку на 3-ем шаге, рассмотрев все возможные остаточные функции, мы не получили новых остаточных функций, то исходная функция f является ограниченно-детерминированной (у нее 4 различных остаточных функции). Следовательно, функция f является автоматной по критерию автоматности.

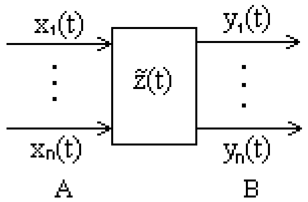


Информационное дерево для этой функции имеет 4 различных вида вершин: $f = f_0$ ○, f_{10} △, f_1 □, f_{11} ◇:

Теперь построим диаграмму Мура, т.е. зададим графически искомый автомат. Для этого “склеим” все одинаковые вершины.



4.2. Автоматы с несколькими входами и выходами

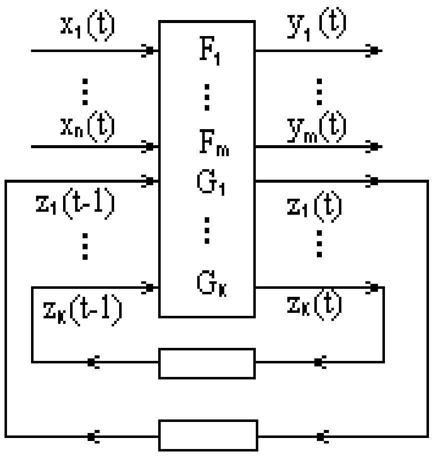


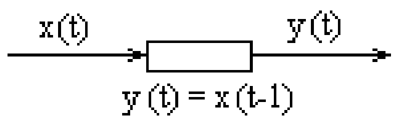
$\widetilde{x}(t) = (x_1(t), \ldots, x_n(t))$,
$\widetilde{y}(t) = (y_1(t), \ldots, y_n(t))$
$\widetilde{z}(t) = (z_1(t), \ldots, z_n(t))$ - вектор-функции,

$$\begin{cases} \widetilde{y}(t) = \widetilde{F}(\widetilde{x}(t), \widetilde{z}(t-1)) \\ \widetilde{z}(t) = \widetilde{G}(\widetilde{x}(t), \widetilde{z}(t-1)) \\ \widetilde{z}(0) = \widetilde{c} \end{cases} \qquad \begin{aligned} \widetilde{F} &= (F_1, \ldots, F_m) \\ \widetilde{G} &= (G_1, \ldots, G_k) \end{aligned} \qquad \begin{aligned} \widetilde{F} &: A^n \times Q^k \to B^m \\ \widetilde{G} &: A^n \times Q^k \to Q^k \end{aligned}$$

F_i: $A^n \times Q^k \to B$ $\quad$ G_j: $A^n \times Q^k \to Q$ $\quad$ $i = 1, \ldots, m$; $j = 1, \ldots, k$

Произвольная синхронная логическая сеть





4.2.1. Реализация сложения с помощью конечного автомата

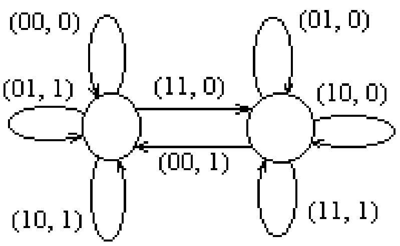
Реализуем сложение натуральных чисел в двоичном виде:

$$\begin{array}{ccccc} & x_1(1) & x_1(2) & \ldots & x_1(n) & 0 \\ + & x_2(1) & x_2(2) & \ldots & x_2(n) & 0 \\ \hline & y(1) & y(2) & \ldots & y(n) & y(n+1) \end{array}$$

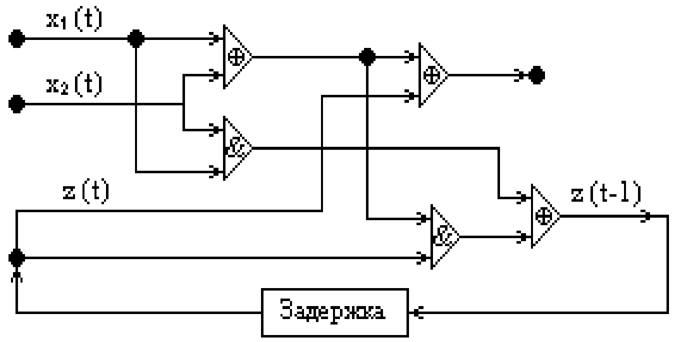
$$z(t)=\begin{cases}1, \text{есть перено с } 1 \text{ в следующий разряд}\\ 0, \text{нет перено са } 1 \text{ в следующий разряд}\end{cases}$$

Уравнения автомата, реализующего сложение: $\begin{cases} y(t)=x_1(t)\oplus x_2(t)\oplus z(t-1)\\ z(t)=x_1(t)x_2(t)\oplus z(t-1)(x_1(t)\oplus x_2(t))\\ z(0)=0\end{cases}$

Диаграмма Мура:



Логическая сеть для данного автомата:



4.2.2. Нереализуемость умножения с помощью конечного автомата

Обозначения: $a^n=\underbrace{a\ldots a}_{\text{n букв}}$ - слово $\qquad a^{\omega} = a \ldots a \ldots$ - сверхслово.

Натуральные числа в двоичной системе имеют вид:

<1> = 1000…
<2> = 0100…
<3> = 1100…
<4> = 0010…

$\Rightarrow$ <2^n> = 0^n10^{ω}

Рассмотрим функцию f – возведение в квадрат: f(<n>) = <n^2>, тогда f(<2^n>) = <2^{2n}> и, значит, $f(0^n10^{\omega})=0^{2n}10^{\omega}$. Но с другой стороны, по 2-му условию детерминированности функции $f(0^n10^{\omega})= f(0^n)f_{0^n}(10^{\omega})$. Из данных равенств следует, что $f_{0^n}(10^{\omega})= 0^n10^{\omega}$, так как $f(0^n) = 0^n$.

Таким образом, вес функции f бесконечен. Значит, по критерию автоматности функция f – не автоматная. Следовательно, нет автомата, реализующего ее.