МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное

образовательное учреждение высшего образования

«ЧЕРЕПОВЕЦКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Институт информационных технологий

Теория автоматов и формальных языков

ЛАБОРАТОРНАЯ РАБОТА № 1-2

МОДЕЛИРОВАНИЕ РАБОТЫ АВТОМАТА МИЛИ

Исполнитель:

Студент 1ПИб-02-3оп-22

Минеев Иван Александрович

Руководитель:

Ганичева Оксана  
Георгиевна

2023 г.

1. Описание работы автомата «Банкомат Сбербанк».

Автомат «Банкомат Сбербанк» показывает экран ожидания карты, до того момента, пока клиент не начнет с ним взаимодействовать. Взаимодействие с автоматом в свою очередь осуществляется путем прикладывания карты банка к специальному считывателю для «бесконтактного входа», либо внесением карты непосредственно внутрь банкомата. После этого банкомат потребует ввести пин-код, для входа в главное меню пользователя. В самом меню клиент может выбрать несколько операций, которые он хочет выполнить со своей картой или счётом. Например, внести или снять наличные, а также оплатить услуги. При завершении операции автомат выходит на начальный экран ожидания карты.

1. Формальное описание модели автомата

|  |  |
| --- | --- |
| **S:** | **Y:** |
| S0 – ожидание карты | y0 – сообщение «Карта считана», отображение экрана ввода пин-кода |
| S1 – принята карта | y1 – сообщение «Карта принята», отображение экрана ввода пин-кода |
| S2 – пин-код принят | у2 – сообщение «Пин-код неверный» |
| S3 – выбрана операция снятия денег | у3 – сообщение «Пин-код верный», отображение главного экрана |
| S4 – выбрана операция внесения денег | у4 – сообщение «Операция выбрана» |
| S5 – выбрана операция оплаты услуг | у5 – отображение главного экрана |
| S6 – введена сумма, которую необходимо снять | у6 – отображение главного экрана |
| S7 – внесена сумма, которую необходимо положить | у7 – отображение главного экрана |
| S8 – выбрана необходимая услуга | у8 – отображение экрана операции |
| S9 – операция подтверждена | у9 – вывод введенной суммы |
| S10 – операция выполнена | у10 – вывод внесенной суммы |
| S11 – карта возвращена | у11 – вывод выбранной услуги |
| **X:** | у12 – вывод результатов операции и баланса |
| x0 – приложена карта (бесконтактный вход) | у13 – отображение главного экрана |
| x1 – вставлена карта (контактный вход) | у14 – сообщение «Заберите карту» |
| x2 – ввод неверного пин-кода | у15 – вывод начального экрана |
| x3 – ввод верного пин-кода | у16 – вывод начального экрана |
| x4 – выбор операции снятия денег |  |
| x5 – выбор операции внесения денег |  |
| x6 – выбор операции оплаты услуг |  |
| x7 – нажата кнопка «Вернуться на главный экран» |  |
| x8 – ввод суммы, которую необходимо снять |  |
| x9 – внос суммы, которую необходимо положить |  |
| x10 – выбор услуги |  |
| x11 – подтверждение операции |  |
| х12 – проведение операции |  |
| х13 – запрос на возврат карты |  |
| х14 – выход из системы |  |

2.1 Функция переходов:

δ: S \* x -> S

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X S | S0 | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | S10 | S11 |
| x0 | S1 |  |  |  |  |  |  |  |  |  |  |  |
| x1 | S1 |  |  |  |  |  |  |  |  |  |  |  |
| x2 |  | S1 |  |  |  |  |  |  |  |  |  |  |
| x3 |  | S2 |  |  |  |  |  |  |  |  |  |  |
| x4 |  |  | S3 |  |  |  |  |  |  |  |  |  |
| x5 |  |  | S4 |  |  |  |  |  |  |  |  |  |
| x6 |  |  | S5 |  |  |  |  |  |  |  |  |  |
| x7 |  |  |  | S2 | S2 | S2 |  |  |  |  | S2 |  |
| x8 |  |  |  | S6 |  |  |  |  |  |  |  |  |
| x9 |  |  |  |  | S7 |  |  |  |  |  |  |  |
| x10 |  |  |  |  |  | S8 |  |  |  |  |  |  |
| x11 |  |  |  |  |  |  | S9 | S9 | S9 |  |  |  |
| x12 |  |  |  |  |  |  |  |  |  | S10 |  |  |
| x13 |  |  | S11 |  |  |  |  |  |  |  |  |  |
| x14 |  |  | S0 |  |  |  |  |  |  |  |  | S0 |

2.2 Функция выводов

λ: S \* x -> y

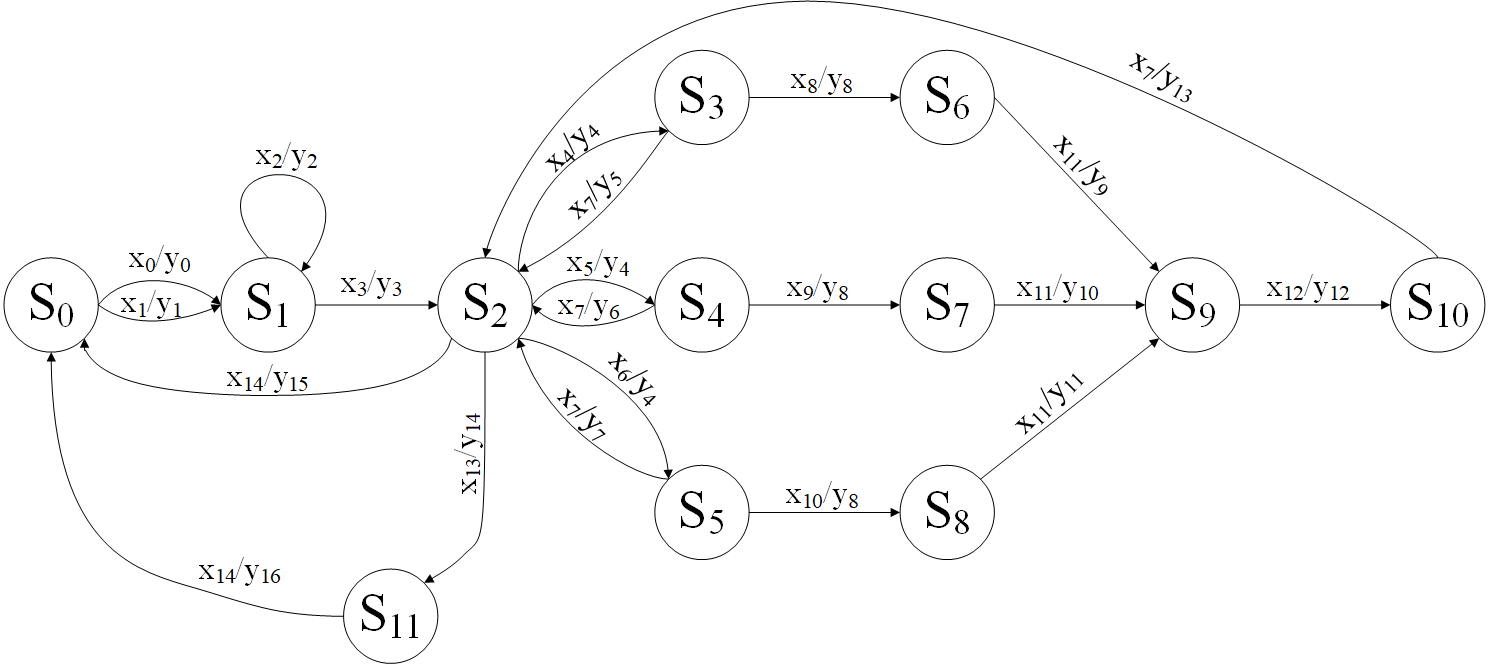
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X S | S0 | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | S10 | S11 |
| x0 | y0 |  |  |  |  |  |  |  |  |  |  |  |
| x1 | y1 |  |  |  |  |  |  |  |  |  |  |  |
| x2 |  | y2 |  |  |  |  |  |  |  |  |  |  |
| x3 |  | y3 |  |  |  |  |  |  |  |  |  |  |
| x4 |  |  | y4 |  |  |  |  |  |  |  |  |  |
| x5 |  |  | y4 |  |  |  |  |  |  |  |  |  |
| x6 |  |  | y4 |  |  |  |  |  |  |  |  |  |
| x7 |  |  |  | y5 | y6 | y7 |  |  |  |  | y13 |  |
| x8 |  |  |  | y8 |  |  |  |  |  |  |  |  |
| x9 |  |  |  |  | y8 |  |  |  |  |  |  |  |
| x10 |  |  |  |  |  | y8 |  |  |  |  |  |  |
| x11 |  |  |  |  |  |  | y9 | y10 | y11 |  |  |  |
| x12 |  |  |  |  |  |  |  |  |  | y12 |  |  |
| x13 |  |  | y14 |  |  |  |  |  |  |  |  |  |
| x14 |  |  | y15 |  |  |  |  |  |  |  |  | y16 |

2.3 Совмещенная функция

δ / λ:

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| X S | S0 | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | S10 | S11 |
| x0 | S1  y0 |  |  |  |  |  |  |  |  |  |  |  |
| x1 | S1  y1 |  |  |  |  |  |  |  |  |  |  |  |
| x2 |  | S1  y2 |  |  |  |  |  |  |  |  |  |  |
| x3 |  | S2  y3 |  |  |  |  |  |  |  |  |  |  |
| x4 |  |  | S3  y4 |  |  |  |  |  |  |  |  |  |
| x5 |  |  | S4  y4 |  |  |  |  |  |  |  |  |  |
| x6 |  |  | S5  y4 |  |  |  |  |  |  |  |  |  |
| x7 |  |  |  | S2  y5 | S2  y6 | S2  y7 |  |  |  |  | S2  y13 |  |
| x8 |  |  |  | S6  y8 |  |  |  |  |  |  |  |  |
| x9 |  |  |  |  | S7  y8 |  |  |  |  |  |  |  |
| x10 |  |  |  |  |  | S8  y8 |  |  |  |  |  |  |
| x11 |  |  |  |  |  |  | S9  y9 | S9  y10 | S9  y11 |  |  |  |
| x12 |  |  |  |  |  |  |  |  |  | S10  y12 |  |  |
| x13 |  |  | S11  y14 |  |  |  |  |  |  |  |  |  |
| x14 |  |  | S0  y15 |  |  |  |  |  |  |  |  | S0  y16 |

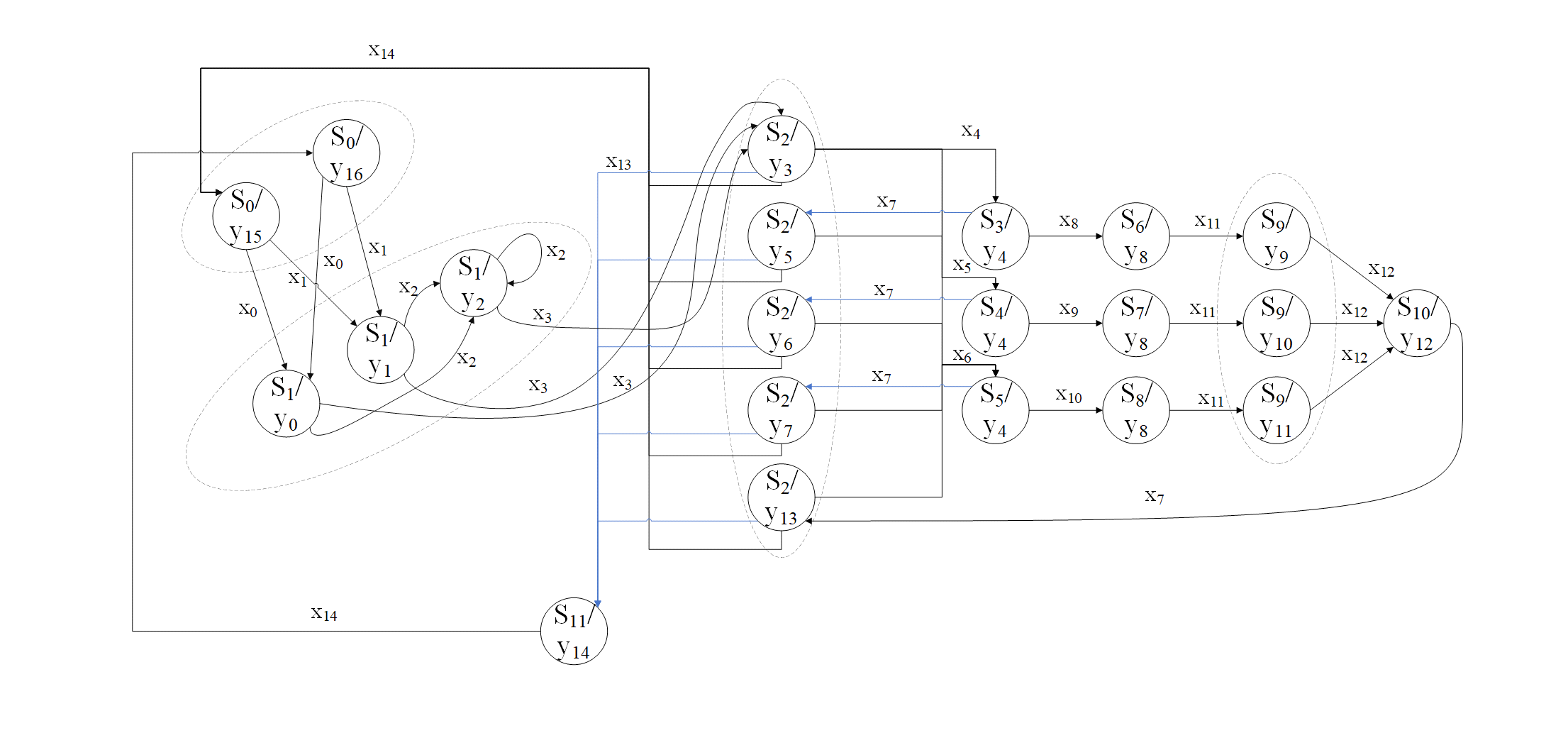
3.1 Граф автомата Мили:



3.2 Протокол работы автомата Мили:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| t | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| X | x1 | x2 | x3 | x6 | x7 | x5 | x7 | x4 | x8 | x11 | x12 | x7 | x5 | x7 | x6 | x10 | x11 | x12 | x7 | x14 |  |
| S | S0 | S1 | S1 | S2 | S5 | S2 | S4 | S2 | S3 | S6 | S9 | S10 | S2 | S4 | S2 | S5 | S8 | S9 | S10 | S2 | S0 |
| Y | y1 | y2 | y3 | y4 | y7 | y4 | y6 | y4 | y8 | y9 | y12 | y13 | y4 | y6 | y4 | y8 | y11 | y12 | y13 | y15 |  |

4.1 Граф автомата Мура

4.2 Автомат Мура. Протокол работы.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| t | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| X | x1 | x2 | x3 | x6 | x7 | x5 | x7 | x4 | x8 | x11 | x12 | x7 | x5 | x7 | x6 | x10 | x11 | x12 | x7 | x14 |  |
| S | S0 | S1 | S1 | S2 | S5 | S2 | S4 | S2 | S3 | S6 | S9 | S10 | S2 | S4 | S2 | S5 | S8 | S9 | S10 | S2 | S0 |
| Y | y1 | y2 | y3 | y4 | y7 | y4 | y6 | y4 | y8 | y9 | y12 | y13 | y4 | y6 | y4 | y8 | y11 | y12 | y13 | y15 |  |

1. Программная реализация  
   Реализация с помощью программного кода:

#pragma once

#include <iostream>

#include <string>

#include <stdlib.h>

#include <string.h>

#include <msclr\marshal.h>

#include <msclr\marshal\_cppstd.h>

#include <msclr\marshal\_windows.h>

#include <msclr\marshal\_atl.h>

namespace Проект1 {

using namespace System;

using namespace System::ComponentModel;

using namespace System::Collections;

using namespace System::Windows::Forms;

using namespace System::Data;

using namespace System::Drawing;

using namespace msclr::interop;

using namespace std;

/// <summary>

/// Сводка для MyForm

/// </summary>

public ref class MyForm : public System::Windows::Forms::Form

{

public:

MyForm(void)

{

InitializeComponent();

//

//TODO: добавьте код конструктора

//

}

protected:

/// <summary>

/// Освободить все используемые ресурсы.

/// </summary>

~MyForm()

{

if (components)

{

delete components;

}

}

protected:

private: System::Windows::Forms::TextBox^ textBox4;

private: System::Windows::Forms::TextBox^ textBox3;

private: System::Windows::Forms::TextBox^ textBox1;

private: System::Windows::Forms::Label^ label3;

private: System::Windows::Forms::Button^ button2;

private: System::Windows::Forms::Label^ label2;

private: System::Windows::Forms::Label^ label1;

private: System::Windows::Forms::TextBox^ textBox2;

private: System::Windows::Forms::Button^ button1;

private: System::Windows::Forms::Label^ label47;

private: System::Windows::Forms::Label^ label48;

private: System::Windows::Forms::Label^ label49;

private: System::Windows::Forms::PictureBox^ pictureBox1;

private: System::Windows::Forms::TextBox^ textBox5;

private: System::Windows::Forms::TextBox^ textBox6;

private: System::Windows::Forms::TextBox^ textBox7;

private: System::Windows::Forms::Label^ label4;

private:

/// <summary>

/// Обязательная переменная конструктора.

/// </summary>

System::ComponentModel::Container^ components;

#pragma region Windows Form Designer generated code

/// <summary>

/// Требуемый метод для поддержки конструктора — не изменяйте

/// содержимое этого метода с помощью редактора кода.

/// </summary>

void InitializeComponent(void)

{

System::ComponentModel::ComponentResourceManager^ resources = (gcnew System::ComponentModel::ComponentResourceManager(MyForm::typeid));

this->textBox4 = (gcnew System::Windows::Forms::TextBox());

this->textBox3 = (gcnew System::Windows::Forms::TextBox());

this->textBox1 = (gcnew System::Windows::Forms::TextBox());

this->label3 = (gcnew System::Windows::Forms::Label());

this->button2 = (gcnew System::Windows::Forms::Button());

this->label2 = (gcnew System::Windows::Forms::Label());

this->label1 = (gcnew System::Windows::Forms::Label());

this->textBox2 = (gcnew System::Windows::Forms::TextBox());

this->button1 = (gcnew System::Windows::Forms::Button());

this->label47 = (gcnew System::Windows::Forms::Label());

this->label48 = (gcnew System::Windows::Forms::Label());

this->label49 = (gcnew System::Windows::Forms::Label());

this->pictureBox1 = (gcnew System::Windows::Forms::PictureBox());

this->textBox5 = (gcnew System::Windows::Forms::TextBox());

this->textBox6 = (gcnew System::Windows::Forms::TextBox());

this->textBox7 = (gcnew System::Windows::Forms::TextBox());

this->label4 = (gcnew System::Windows::Forms::Label());

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->pictureBox1))->BeginInit();

this->SuspendLayout();

//

// textBox4

//

this->textBox4->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14));

this->textBox4->Location = System::Drawing::Point(56, 541);

this->textBox4->Name = L"textBox4";

this->textBox4->ReadOnly = true;

this->textBox4->Size = System::Drawing::Size(614, 29);

this->textBox4->TabIndex = 21;

//

// textBox3

//

this->textBox3->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14));

this->textBox3->Location = System::Drawing::Point(56, 611);

this->textBox3->Name = L"textBox3";

this->textBox3->ReadOnly = true;

this->textBox3->Size = System::Drawing::Size(614, 29);

this->textBox3->TabIndex = 20;

//

// textBox1

//

this->textBox1->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 14));

this->textBox1->Location = System::Drawing::Point(56, 576);

this->textBox1->Name = L"textBox1";

this->textBox1->ReadOnly = true;

this->textBox1->Size = System::Drawing::Size(614, 29);

this->textBox1->TabIndex = 19;

//

// label3

//

this->label3->AutoSize = true;

this->label3->Font = (gcnew System::Drawing::Font(L"Times New Roman", 15.75F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label3->ForeColor = System::Drawing::SystemColors::ControlText;

this->label3->Location = System::Drawing::Point(77, 363);

this->label3->Name = L"label3";

this->label3->Size = System::Drawing::Size(444, 23);

this->label3->TabIndex = 18;

this->label3->Text = L"(числа через один пробел, без запятых и буквы х)";

//

// button2

//

this->button2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 15.75F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button2->ForeColor = System::Drawing::SystemColors::ControlText;

this->button2->Location = System::Drawing::Point(353, 424);

this->button2->Name = L"button2";

this->button2->Size = System::Drawing::Size(273, 67);

this->button2->TabIndex = 17;

this->button2->Text = L"Автоматический режим";

this->button2->UseVisualStyleBackColor = true;

this->button2->Click += gcnew System::EventHandler(this, &MyForm::button2\_Click);

//

// label2

//

this->label2->AutoSize = true;

this->label2->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));

this->label2->ForeColor = System::Drawing::SystemColors::ControlText;

this->label2->Location = System::Drawing::Point(224, 507);

this->label2->Name = L"label2";

this->label2->Size = System::Drawing::Size(233, 31);

this->label2->TabIndex = 16;

this->label2->Text = L"Протокол работы";

//

// label1

//

this->label1->AutoSize = true;

this->label1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 20.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label1->ForeColor = System::Drawing::SystemColors::ControlText;

this->label1->Location = System::Drawing::Point(75, 332);

this->label1->Name = L"label1";

this->label1->Size = System::Drawing::Size(451, 31);

this->label1->TabIndex = 15;

this->label1->Text = L"Введите входную последовательность";

//

// textBox2

//

this->textBox2->Font = (gcnew System::Drawing::Font(L"Times New Roman", 14.25F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox2->Location = System::Drawing::Point(12, 389);

this->textBox2->Name = L"textBox2";

this->textBox2->Size = System::Drawing::Size(614, 29);

this->textBox2->TabIndex = 14;

this->textBox2->TextChanged += gcnew System::EventHandler(this, &MyForm::textBox2\_TextChanged);

//

// button1

//

this->button1->Font = (gcnew System::Drawing::Font(L"Times New Roman", 15.75F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->button1->ForeColor = System::Drawing::SystemColors::ControlText;

this->button1->Location = System::Drawing::Point(12, 424);

this->button1->Name = L"button1";

this->button1->Size = System::Drawing::Size(273, 67);

this->button1->TabIndex = 13;

this->button1->Text = L"Пошаговый режим";

this->button1->UseVisualStyleBackColor = true;

this->button1->Click += gcnew System::EventHandler(this, &MyForm::button1\_Click);

//

// label47

//

this->label47->AutoSize = true;

this->label47->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));

this->label47->ForeColor = System::Drawing::Color::Black;

this->label47->Location = System::Drawing::Point(10, 541);

this->label47->Name = L"label47";

this->label47->Size = System::Drawing::Size(40, 31);

this->label47->TabIndex = 66;

this->label47->Text = L"Х:";

//

// label48

//

this->label48->AutoSize = true;

this->label48->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));

this->label48->ForeColor = System::Drawing::SystemColors::ControlText;

this->label48->Location = System::Drawing::Point(10, 576);

this->label48->Name = L"label48";

this->label48->Size = System::Drawing::Size(40, 31);

this->label48->TabIndex = 67;

this->label48->Text = L"Y:";

//

// label49

//

this->label49->AutoSize = true;

this->label49->Font = (gcnew System::Drawing::Font(L"Microsoft Sans Serif", 20));

this->label49->ForeColor = System::Drawing::SystemColors::ControlText;

this->label49->Location = System::Drawing::Point(10, 609);

this->label49->Name = L"label49";

this->label49->Size = System::Drawing::Size(40, 31);

this->label49->TabIndex = 68;

this->label49->Text = L"S:";

//

// pictureBox1

//

this->pictureBox1->Image = (cli::safe\_cast<System::Drawing::Image^>(resources->GetObject(L"pictureBox1.Image")));

this->pictureBox1->Location = System::Drawing::Point(12, 45);

this->pictureBox1->Margin = System::Windows::Forms::Padding(2);

this->pictureBox1->Name = L"pictureBox1";

this->pictureBox1->Size = System::Drawing::Size(616, 285);

this->pictureBox1->TabIndex = 69;

this->pictureBox1->TabStop = false;

//

// textBox5

//

this->textBox5->Font = (gcnew System::Drawing::Font(L"Times New Roman", 9.75F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox5->Location = System::Drawing::Point(636, 8);

this->textBox5->Multiline = true;

this->textBox5->Name = L"textBox5";

this->textBox5->Size = System::Drawing::Size(184, 355);

this->textBox5->TabIndex = 70;

this->textBox5->Text = resources->GetString(L"textBox5.Text");

this->textBox5->TextChanged += gcnew System::EventHandler(this, &MyForm::textBox5\_TextChanged);

//

// textBox6

//

this->textBox6->Font = (gcnew System::Drawing::Font(L"Times New Roman", 9.75F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox6->Location = System::Drawing::Point(826, 8);

this->textBox6->Multiline = true;

this->textBox6->Name = L"textBox6";

this->textBox6->Size = System::Drawing::Size(229, 355);

this->textBox6->TabIndex = 71;

this->textBox6->Text = resources->GetString(L"textBox6.Text");

//

// textBox7

//

this->textBox7->Font = (gcnew System::Drawing::Font(L"Times New Roman", 9.75F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->textBox7->Location = System::Drawing::Point(1061, 8);

this->textBox7->Multiline = true;

this->textBox7->Name = L"textBox7";

this->textBox7->Size = System::Drawing::Size(180, 355);

this->textBox7->TabIndex = 72;

this->textBox7->Text = resources->GetString(L"textBox7.Text");

//

// label4

//

this->label4->Font = (gcnew System::Drawing::Font(L"Times New Roman", 15.75F, System::Drawing::FontStyle::Regular, System::Drawing::GraphicsUnit::Point,

static\_cast<System::Byte>(204)));

this->label4->Location = System::Drawing::Point(12, 9);

this->label4->Name = L"label4";

this->label4->Size = System::Drawing::Size(196, 29);

this->label4->TabIndex = 73;

this->label4->Text = L"Граф автомата Мили";

//

// MyForm

//

this->AutoScaleDimensions = System::Drawing::SizeF(6, 13);

this->AutoScaleMode = System::Windows::Forms::AutoScaleMode::Font;

this->ClientSize = System::Drawing::Size(1251, 658);

this->Controls->Add(this->label4);

this->Controls->Add(this->textBox7);

this->Controls->Add(this->textBox6);

this->Controls->Add(this->textBox5);

this->Controls->Add(this->pictureBox1);

this->Controls->Add(this->label49);

this->Controls->Add(this->label48);

this->Controls->Add(this->label47);

this->Controls->Add(this->textBox4);

this->Controls->Add(this->textBox3);

this->Controls->Add(this->textBox1);

this->Controls->Add(this->label3);

this->Controls->Add(this->button2);

this->Controls->Add(this->label2);

this->Controls->Add(this->label1);

this->Controls->Add(this->textBox2);

this->Controls->Add(this->button1);

this->FormBorderStyle = System::Windows::Forms::FormBorderStyle::FixedSingle;

this->MaximizeBox = false;

this->Name = L"MyForm";

this->Text = L"Автомат Мили";

this->Load += gcnew System::EventHandler(this, &MyForm::MyForm\_Load);

(cli::safe\_cast<System::ComponentModel::ISupportInitialize^>(this->pictureBox1))->EndInit();

this->ResumeLayout(false);

this->PerformLayout();

}

void Mili(int x[], int y[], int s[], int i, int j, int& sum) {

switch (s[i]) {

case 0:

{

if (s[i] == 0) {

if (x[i] == 0) {

s[i + 1] = 1;

y[i] = 0;

}

else if (x[i] == 1) {

s[i + 1] = 1;

y[i] = 1;

}

else sum++;

}

}

case 1:

{

if (s[i] == 1) {

if (x[i] == 2) {

s[i + 1] = 1;

y[i] = 2;

}

else if (x[i] == 3) {

s[i + 1] = 2;

y[i] = 3;

}

else sum++;

}

}

case 2:

{

if (s[i] == 2) {

if (x[i] == 4) {

s[i + 1] = 3;

y[i] = 4;

}

else if (x[i] == 5) {

s[i + 1] = 4;

y[i] = 4;

}

else if (x[i] == 6) {

s[i + 1] = 5;

y[i] = 4;

}

else if (x[i] == 13) {

s[i + 1] = 11;

y[i] = 14;

}

else if (x[i] == 14) {

s[i + 1] = 0;

y[i] = 15;

}

else sum++;

}

}

case 3:

{

if (s[i] == 3) {

if (x[i] == 7) {

s[i + 1] = 2;

y[i] = 5;

}

else if (x[i] == 8) {

s[i + 1] = 6;

y[i] = 8;

}

else sum++;

}

}

case 4:

{

if (s[i] == 4) {

if (x[i] == 7) {

s[i + 1] = 2;

y[i] = 6;

}

else if (x[i] == 9) {

s[i + 1] = 7;

y[i] = 8;

}

else sum++;

}

}

case 5:

{

if (s[i] == 5) {

if (x[i] == 7) {

s[i + 1] = 2;

y[i] = 7;

}

else if (x[i] == 10) {

s[i + 1] = 8;

y[i] = 8;

}

else sum++;

}

}

case 6:

{

if (s[i] == 6) {

if (x[i] == 11) {

s[i + 1] = 9;

y[i] = 9;

}

else sum++;

}

}

case 7:

{

if (s[i] == 7) {

if (x[i] == 11) {

s[i + 1] = 9;

y[i] = 10;

}

else sum++;

}

}

case 8:

{

if (s[i] == 8) {

if (x[i] == 11) {

s[i + 1] = 9;

y[i] = 11;

}

else sum++;

}

}

case 9:

{

if (s[i] == 9) {

if (x[i] == 12) {

s[i + 1] = 10;

y[i] = 12;

}

else sum++;

}

}

case 10:

{

if (s[i] == 10) {

if (x[i] == 7) {

s[i + 1] = 2;

y[i] = 13;

}

else sum++;

}

}

case 11:

{

if (s[i] == 11) {

if (x[i] == 14) {

s[i + 1] = 0;

y[i] = 16;

}

else sum++;

}

}

}

}

#pragma endregion

int click = 0;

private: System::Void button1\_Click(System::Object^ sender, System::EventArgs^ e) {

setlocale(LC\_ALL, "Russian");

int n = 0;

String^ sx = textBox2->Text;

string str\_x = marshal\_as<string>(sx), str\_y = "", str\_s = "";

for (int i = 0; i < str\_x.length(); i++)

if (i != 0)

if ((str\_x[i] == ' ' || str\_x[i] == '\0') && str\_x[i - 1] != ' ') n++;

int f = 0, j = 0, sum = 0;

int\* x = new int[n + 1];

int\* y = new int[n + 1];

int\* s = new int[n + 2];

for (int i = 0; i < str\_x.length(); i++) {

f \*= 10;

if (str\_x[i] == '0' && sum == 0) f = f;

else if (str\_x[i] == '1' && sum == 0) f++;

else if (str\_x[i] == '2' && sum == 0) f += 2;

else if (str\_x[i] == '3' && sum == 0) f += 3;

else if (str\_x[i] == '4' && sum == 0) f += 4;

else if (str\_x[i] == '5' && sum == 0) f += 5;

else if (str\_x[i] == '6' && sum == 0) f += 6;

else if (str\_x[i] == '7' && sum == 0) f += 7;

else if (str\_x[i] == '8' && sum == 0) f += 8;

else if (str\_x[i] == '9' && sum == 0) f += 9;

else if (str\_x[i] == ' ' && sum == 0 && i != 0) {

if (str\_x[i - 1] == ' ') {

if (sum == 0) sum++;

}

if (str\_x[i + 1] != '\0') {

f /= 10;

x[j] = f;

j++;

f = 0;

}

else f /= 10;

}

else {

if (sum == 0) sum++;

}

}

x[j] = f;

s[0] = 0;

if (sum == 0 && click <= j)

for (int i = 0; i <= click; i++) {

Mili(x, y, s, i, click, sum);

}

else sum++;

if (sum == 0) {

for (int i = 0; i <= j; i++) {

if (s[i] == 0) str\_s += "0 ";

else if (s[i] == 1) str\_s += "1 ";

else if (s[i] == 2) str\_s += "2 ";

else if (s[i] == 3) str\_s += "3 ";

else if (s[i] == 4) str\_s += "4 ";

else if (s[i] == 5) str\_s += "5 ";

else if (s[i] == 6) str\_s += "6 ";

else if (s[i] == 7) str\_s += "7 ";

else if (s[i] == 8) str\_s += "8 ";

else if (s[i] == 9) str\_s += "9 ";

else if (s[i] == 10) str\_s += "10 ";

else if (s[i] == 11) str\_s += "11 ";

}

for (int i = 0; i <= j; i++) {

if (y[i] == 0) str\_y += "0 ";

else if (y[i] == 1) str\_y += "1 ";

else if (y[i] == 2) str\_y += "2 ";

else if (y[i] == 3) str\_y += "3 ";

else if (y[i] == 4) str\_y += "4 ";

else if (y[i] == 5) str\_y += "5 ";

else if (y[i] == 6) str\_y += "6 ";

else if (y[i] == 7) str\_y += "7 ";

else if (y[i] == 8) str\_y += "8 ";

else if (y[i] == 9) str\_y += "9 ";

else if (y[i] == 10) str\_y += "10 ";

else if (y[i] == 11) str\_y += "11 ";

else if (y[i] == 12) str\_y += "12 ";

else if (y[i] == 13) str\_y += "13 ";

else if (y[i] == 14) str\_y += "14 ";

else if (y[i] == 15) str\_y += "15 ";

else if (y[i] == 16) str\_y += "16 ";

}

}

String^ ss = marshal\_as<String^>(str\_s);

String^ sy = marshal\_as<String^>(str\_y);

if (sum == 0) {

textBox4->Text = sx;

textBox1->Text = sy;

textBox3->Text = ss;

}

else {

textBox4->Text = "Ошибка!";

textBox1->Text = "";

textBox3->Text = "";

}

click++;

delete[] x;

delete[] y;

delete[] s;

}

private: System::Void button2\_Click(System::Object^ sender, System::EventArgs^ e) {

setlocale(LC\_ALL, "Russian");

int n = 0;

String^ sx = textBox2->Text;

string str\_x = marshal\_as<string>(sx), str\_y = "", str\_s = "";

for (int i = 0; i < str\_x.length(); i++)

if (i != 0)

if ((str\_x[i] == ' ' || str\_x[i] == '\0') && str\_x[i - 1] != ' ') n++;

int f = 0, j = 0, sum = 0;

int\* x = new int[n + 1];

int\* y = new int[n + 1];

int\* s = new int[n + 2];

for (int i = 0; i < str\_x.length(); i++) {

f \*= 10;

if (str\_x[i] == '0' && sum == 0) f = f;

else if (str\_x[i] == '1' && sum == 0) f++;

else if (str\_x[i] == '2' && sum == 0) f += 2;

else if (str\_x[i] == '3' && sum == 0) f += 3;

else if (str\_x[i] == '4' && sum == 0) f += 4;

else if (str\_x[i] == '5' && sum == 0) f += 5;

else if (str\_x[i] == '6' && sum == 0) f += 6;

else if (str\_x[i] == '7' && sum == 0) f += 7;

else if (str\_x[i] == '8' && sum == 0) f += 8;

else if (str\_x[i] == '9' && sum == 0) f += 9;

else if (str\_x[i] == ' ' && sum == 0 && i != 0) {

if (str\_x[i - 1] == ' ') {

if (sum == 0) sum++;

}

if (str\_x[i + 1] != '\0') {

f /= 10;

x[j] = f;

j++;

f = 0;

}

else f /= 10;

}

else {

if (sum == 0) sum++;

}

}

x[j] = f;

s[0] = 0;

if (sum == 0)

for (int i = 0; i <= j; i++) {

Mili(x, y, s, i, j, sum);

}

if (sum == 0) {

for (int i = 0; i <= j + 1; i++) {

if (s[i] == 0) str\_s += "0 ";

else if (s[i] == 1) str\_s += "1 ";

else if (s[i] == 2) str\_s += "2 ";

else if (s[i] == 3) str\_s += "3 ";

else if (s[i] == 4) str\_s += "4 ";

else if (s[i] == 5) str\_s += "5 ";

else if (s[i] == 6) str\_s += "6 ";

else if (s[i] == 7) str\_s += "7 ";

else if (s[i] == 8) str\_s += "8 ";

else if (s[i] == 9) str\_s += "9 ";

else if (s[i] == 10) str\_s += "10 ";

else if (s[i] == 11) str\_s += "11 ";

}

for (int i = 0; i <= j; i++) {

if (y[i] == 0) str\_y += "0 ";

else if (y[i] == 1) str\_y += "1 ";

else if (y[i] == 2) str\_y += "2 ";

else if (y[i] == 3) str\_y += "3 ";

else if (y[i] == 4) str\_y += "4 ";

else if (y[i] == 5) str\_y += "5 ";

else if (y[i] == 6) str\_y += "6 ";

else if (y[i] == 7) str\_y += "7 ";

else if (y[i] == 8) str\_y += "8 ";

else if (y[i] == 9) str\_y += "9 ";

else if (y[i] == 10) str\_y += "10 ";

else if (y[i] == 11) str\_y += "11 ";

else if (y[i] == 12) str\_y += "12 ";

else if (y[i] == 13) str\_y += "13 ";

else if (y[i] == 14) str\_y += "14 ";

else if (y[i] == 15) str\_y += "15 ";

else if (y[i] == 16) str\_y += "16 ";

}

}

String^ ss = marshal\_as<String^>(str\_s);

String^ sy = marshal\_as<String^>(str\_y);

if (sum == 0) {

textBox4->Text = sx;

textBox1->Text = sy;

textBox3->Text = ss;

}

else {

textBox4->Text = "Ошибка!";

textBox1->Text = "";

textBox3->Text = "";

}

delete[] x;

delete[] y;

delete[] s;

}

private: System::Void label27\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void textBox2\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void MyForm\_Load(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void pictureBox1\_Click(System::Object^ sender, System::EventArgs^ e) {

}

private: System::Void textBox5\_TextChanged(System::Object^ sender, System::EventArgs^ e) {

}

};

}

Программная модель представлена на рис. 1.

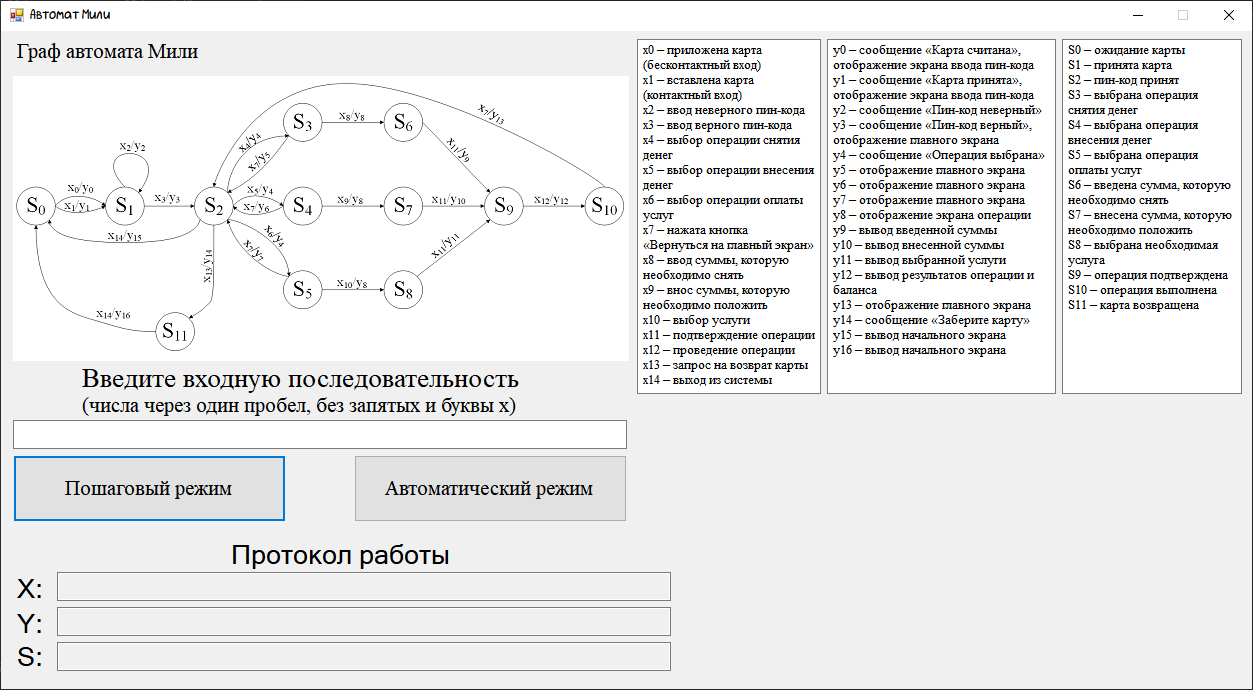


Рис. 1.

1. Вывод

В ходе выполнения работы были описаны и построены автоматы Мили и Мура. Основными характеристиками разобранного автомата Мили являются детерминированность, асинхронность, абстрактность, частичная определенность. Была создана программа, выполняющая работу на основе построенного графа Мили для автомата «Банкомат Сбербанка».