БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

Ханты-Мансийского автономного округа – Югры

«Сургутский государственный университет»

Политехнический институт Кафедра автоматики и компьютерных систем

ОТЧЕТ

по лабораторной работе

дисциплина Теория Вычислительных Процессов

Тема: Модель Светофора с Кнопкой

Выполнил: студент группы 609-21

Яковлев Алексей Викторович

Принял: доцент кафедры АиКС

Брагинский М. Я.

Сургут 2025 г.

**Задание:**

1. Описать работу ЦА и программную модель ЦА, управляющего двухсекционным светофором с кнопкой и индикатором ожидания

**Граф работы (рисунок 1):**

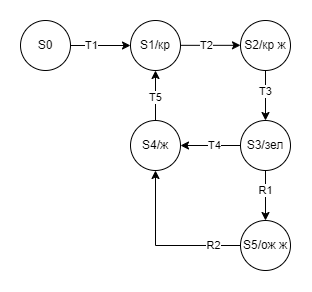


Рисунок 1 - Граф работы ЦА.

**Таблица переходов (таблица 1):**

Таблица 1 – Таблица переходов ЦА

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Входные Сигналы** | **Состояния** | | | | | |
| S0 | S1 | S2 | S3 | S4 | S5 |
| T1 | S1 | - | - | - | - | - |
| T2 | - | S2 | - | - | - | - |
| T3 | - | - | S3 | - | - | - |
| T4 | - | - | - | S4 | - | - |
| T5 | - | - | - | - | S1 | - |
| R1 | - | - | - | S5 | - | - |
| R2 | - | - | - | - | - | S4 |

**Входные сигналы:**

T1 – Светофор был запущен через 3 секунды;

T2 – Таймер насчитал 10 секунд;

T3 – Таймер насчитал 3 секунды;

T4 – Таймер насчитал 10 секунд;

T5 – Таймер насчитал 3 секунды;

R1 – Была нажата кнопка запроса на переход;

R2 – Прошло 3 секунды после нажатия кнопки запроса на переход.

**Состояния:**

S0 – Светофор выключен, не горит ни один сигнал светофора;

S1 – Горит красный сигнал светофора;

S2 – Горят красный и желтый сигналы светофора;

S3 – Горит зеленый сигнал светофора;

S4 – Горит желтый сигнал светофора;

S5 – Горит зеленый сигнал светофора и индикатор ожидания перехода.

**Код программного обеспечения (листинг 1):**

Код был написан на языке программирования C# c MAUI в среде разработки Visual Studio 2022.

Листинг 1. Код программного обеспечения:

using System.Timers;

namespace lab1.\_1

{

public class TrafficLightController

{

private enum LightState {Off, Red, RedYellow, Green, Yellow , WaitYellow }

private LightState currentState = LightState.Off;

private int timeLeft;

private readonly System.Timers.Timer timer;

private readonly BoxView red1, yellow1, green1, pedRed1, pedGreen1;

private readonly Label timerLabel1;

private readonly BoxView red2, yellow2, green2, pedRed2, pedGreen2;

private readonly Label timerLabel2;

private readonly Button forceYellowButton;

private readonly Button forceYellowButton2;

public TrafficLightController(

BoxView red1, BoxView yellow1, BoxView green1, Label timerLabel1, BoxView pedRed1, BoxView pedGreen1,

BoxView red2, BoxView yellow2, BoxView green2, Label timerLabel2, BoxView pedRed2, BoxView pedGreen2,

Button forceYellowButton, Button forceYellowButton2)

{

this.red1 = red1;

this.yellow1 = yellow1;

this.green1 = green1;

this.timerLabel1 = timerLabel1;

this.pedRed1 = pedRed1;

this.pedGreen1 = pedGreen1;

this.red2 = red2;

this.yellow2 = yellow2;

this.green2 = green2;

this.timerLabel2 = timerLabel2;

this.pedRed2 = pedRed2;

this.pedGreen2 = pedGreen2;

this.forceYellowButton = forceYellowButton;

this.forceYellowButton2 = forceYellowButton2;

timer = new System.Timers.Timer(1000);

timer.Elapsed += OnTimedEvent;

timer.AutoReset = true;

}

public void Start()

{

SetLightState(LightState.Off, 3);

timer.Start();

}

private void OnTimedEvent(object sender, ElapsedEventArgs e)

{

if (timeLeft > 0)

{

timeLeft--;

UpdateUI();

}

else

{

switch (currentState)

{

case LightState.Off:

SetLightState(LightState.Red, 9);

break;

case LightState.Red:

SetLightState(LightState.RedYellow, 2);

break;

case LightState.RedYellow:

SetLightState(LightState.Green, 9);

break;

case LightState.Green:

SetLightState(LightState.Yellow, 2);

break;

case LightState.Yellow:

SetLightState(LightState.Red, 9);

break;

case LightState.WaitYellow:

SetLightState(LightState.Yellow, 2);

break;

}

}

}

private void SetLightState(LightState state, int duration)

{

currentState = state;

timeLeft = duration;

UpdateUI();

forceYellowButton.Dispatcher.Dispatch(() =>

{

// Разблокируем кнопку, когда светофор становится красным

if (state == LightState.Red)

{

forceYellowButton.Text = "Нажмите";

forceYellowButton.IsEnabled = true;

forceYellowButton2.Text = "Нажмите";

forceYellowButton2.IsEnabled = true;

}

});

}

private void UpdateUI()

{

void UpdateTrafficLight(BoxView red, BoxView yellow, BoxView green, Label timerLabel, BoxView pedRed, BoxView pedGreen)

{

red.Color = (currentState == LightState.Red || currentState == LightState.RedYellow) ? Colors.Red : Colors.Gray;

yellow.Color = (currentState == LightState.Yellow || currentState == LightState.RedYellow) ? Colors.Yellow : Colors.Gray;

green.Color = (currentState == LightState.Green || currentState == LightState.WaitYellow) ? Colors.Green : Colors.Gray;

pedRed.Color = (currentState == LightState.Green || currentState == LightState.Yellow || currentState == LightState.RedYellow || currentState == LightState.WaitYellow) ? Colors.Red : Colors.Gray;

pedGreen.Color = (currentState == LightState.Red) ? Colors.Green : Colors.Gray;

timerLabel.Dispatcher.Dispatch(() => timerLabel.Text = timeLeft.ToString());

}

UpdateTrafficLight(red1, yellow1, green1, timerLabel1, pedRed1, pedGreen1);

UpdateTrafficLight(red2, yellow2, green2, timerLabel2, pedRed2, pedGreen2);

}

public void ForceYellow()

{

if (currentState == LightState.Green)

{

forceYellowButton.Text = "Ожидайте";

forceYellowButton.IsEnabled = false;

forceYellowButton2.Text = "Ожидайте";

forceYellowButton2.IsEnabled = false;

SetLightState(LightState.WaitYellow, 3);

}

}

}

}  
  
Листинг 2. Код программного интерфейса:

<?xml version="1.0" encoding="utf-8" ?>

<ContentPage xmlns="http://schemas.microsoft.com/dotnet/2021/maui"

xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"

x:Class="lab1.\_1.MainPage">

<HorizontalStackLayout HorizontalOptions="Center" Spacing="30">

<!-- Первый светофор -->

<VerticalStackLayout x:Name="TrafficLight1" VerticalOptions="Center" HorizontalOptions="Center" Spacing="10">

<BoxView x:Name="RedLight1" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<BoxView x:Name="YellowLight1" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<BoxView x:Name="GreenLight1" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<Label x:Name="TimerLabel1" FontSize="24" HorizontalOptions="Center" />

</VerticalStackLayout>

<VerticalStackLayout VerticalOptions="Center" HorizontalOptions="Center" Spacing="10" >

<BoxView x:Name="PedestrianRed1" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<BoxView x:Name="PedestrianGreen1" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<Button x:Name="ForceYellowButton" Text="Ждите" Clicked="OnForceYellowButtonClicked"/>

</VerticalStackLayout>

<BoxView WidthRequest="200" BackgroundColor="Transparent"></BoxView>

<!-- Второй светофор -->

<VerticalStackLayout x:Name="TrafficLight2" VerticalOptions="Center" HorizontalOptions="Center" Spacing="10">

<BoxView x:Name="RedLight2" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<BoxView x:Name="YellowLight2" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<BoxView x:Name="GreenLight2" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<Label x:Name="TimerLabel2" FontSize="24" HorizontalOptions="Center" />

</VerticalStackLayout>

<VerticalStackLayout VerticalOptions="Center" HorizontalOptions="Center" Spacing="10" >

<BoxView x:Name="PedestrianRed2" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<BoxView x:Name="PedestrianGreen2" Color="Gray" WidthRequest="50" HeightRequest="50" CornerRadius="25" />

<Button x:Name="ForceYellowButton2" Text="Ждите" Clicked="OnForceYellowButtonClicked"/>

</VerticalStackLayout>

</HorizontalStackLayout>

</ContentPage>

Листинг 3. Код программного обеспечения:

namespace lab1.\_1

{

public partial class MainPage : ContentPage

{

private TrafficLightController trafficLightController;

public MainPage()

{

InitializeComponent();

// Создаем контроллер и передаем в него элементы для обоих светофоров

trafficLightController = new TrafficLightController(

RedLight1, YellowLight1, GreenLight1, TimerLabel1, PedestrianRed1, PedestrianGreen1,

RedLight2, YellowLight2, GreenLight2, TimerLabel2, PedestrianRed2, PedestrianGreen2,

ForceYellowButton, ForceYellowButton2

);

// Запускаем светофор

trafficLightController.Start();

}

private void OnForceYellowButtonClicked(object sender, EventArgs e)

{

trafficLightController.ForceYellow();

}

}

}

**Пример работы программы (рисунки 2-7):**



Рисунок 2 – Светофор в состоянии S0

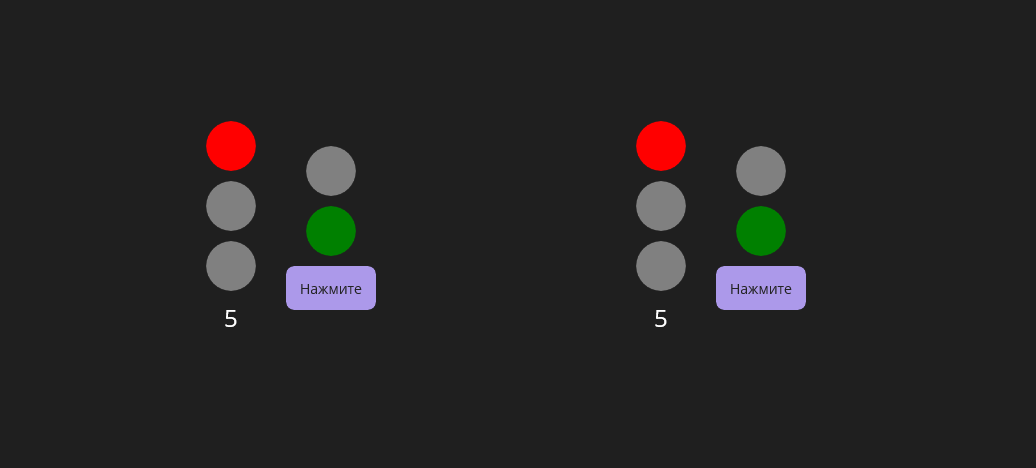


Рисунок 3 – Светофор в состоянии S1

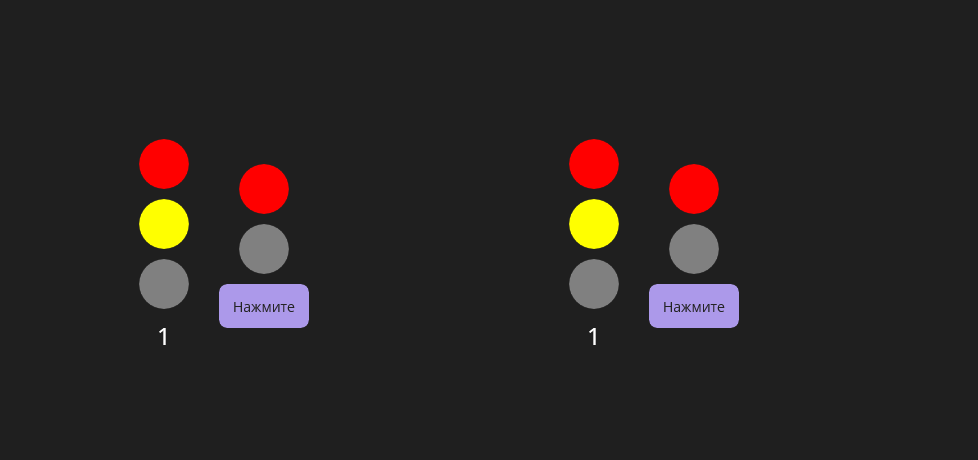


Рисунок 4 – Светофор в состоянии S2

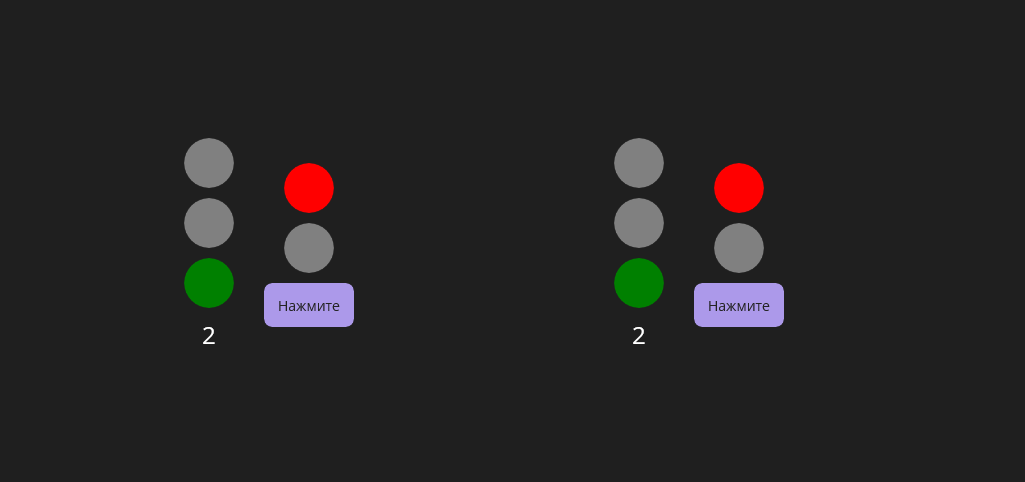


Рисунок 5 – Светофор в состоянии S3

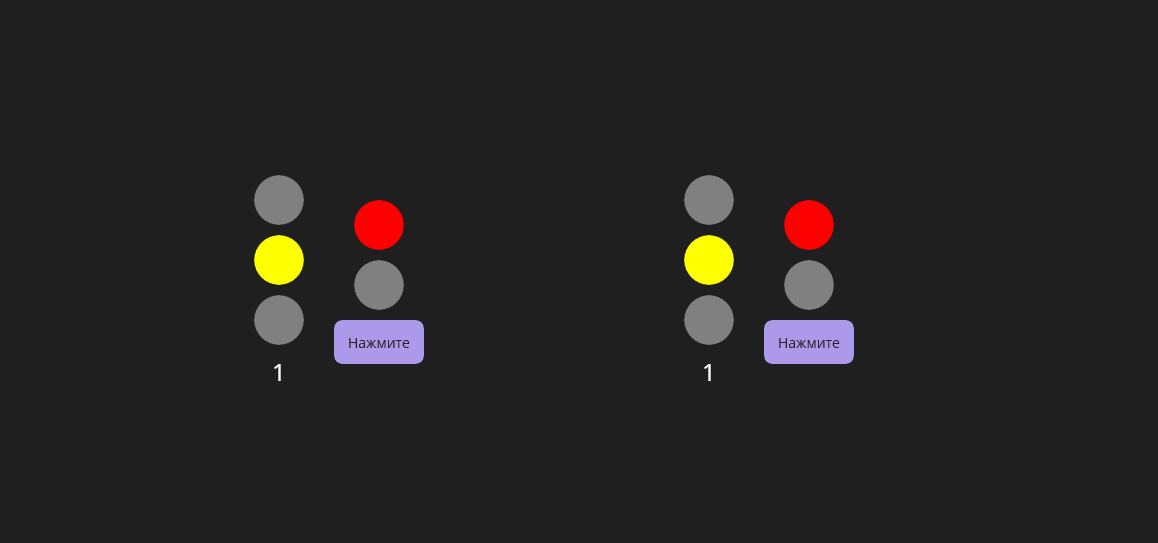


Рисунок 6 – Светофор в состоянии S4



Рисунок 7 – Светофор в состоянии S5

Выводы: в ходе лабораторной работы была создана программная реализация цифрового автомата модели светофора.