

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ "МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ИМЕНИ М.В.ЛОМОНОСОВА"

Физический факультет

Основы Web-технологий

Малышев Владислав Денисович

Визуализация явления двойного лучепреломления с помощью
языка JavaScript

Курсовая работа

Преподаватель:

Алексеев А. А.

Москва — 2016

Оглавление

Введение	3
1 О физических явлениях и инструментах программирования, используемых в работе	4
1.1 Явление двойного лучепреломления	4
1.2 О языке JavaScript	5
1.3 Canvas	6
2 О самой программе	7
2.1 Возможности	7
2.2 Прорисовка объектов	9
2.3 Вставка JavaScript в код HTML	19
Итоги работы	22
Литература	23

Введение

Данная работа посвящена визуализации процесса двойного лучепреломления. Её целью было использование полученных в курсе "Основы Web-технологий" знаний и навыков, более глубокое ознакомление с методами визуализации с помощью языка JavaScript и разработка программы для наглядной визуализации данного явления и дальнейшего использования в преподавательских целях.

Глава 1

О физических явлениях и инструментах программирования, используемых в работе

1.1 Явление двойного лучепреломления

Почти все прозрачные кристаллы диэлектрически оптически анизотропны, то есть оптические свойства света при прохождении через них зависят от направления распространения. Вследствие этого возникает явление, называемое двойным лучепреломлением. Оно заключается в том, что падающий на кристалл пучок света разделяется на два пучка, распространяющихся, вообще говоря, в разных направлениях и с разными скоростями. Физическая природа анизотропии связана с особенностями строения его молекул или особенностями самой кристаллической решётки, в узлах которой находятся атомы или ионы.

Существуют двуосные и одноосные кристаллы. У одноосных кристаллов один из обыкновенных пучков подчиняется обычному закону преломления. Его называют обыкновенным и обозначают индексом o . Другой пучок — необыкновенный (e) — не подчиняется обычному закону преломления, и даже при нормальном падении светового пучка на поверхность кристалла необыкновенный пучок может отклониться от нормали (Рис.1.1).

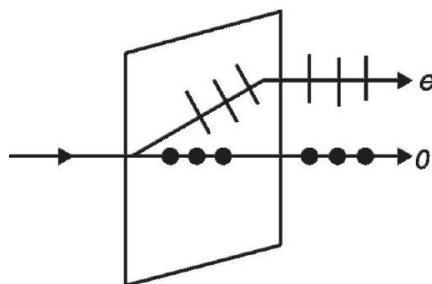


Рис. 1.1: Схематический вид распространения обыкновенного и необыкновенного лучей в одноосном кристалле.

У одноосных кристаллов имеется направление — оптическая ось, вдоль которого обыкновенная и необыкновенная волны распространяются, не разделяясь пространственно и с одинаковой скоростью. Любую плоскость, проходящую через оптическую ось, называют главным сечением кристалла. Обыкновенная и необыкновенная волны линейно поляризованы, причём колебания вектора E в обыкновенной волне совершаются в направлении, перпендикулярном главному сечению кристалла. Колебания же вектора E в необыкновенной волне совершаются в главном сечении кристалла [4].

В данной программе рассматривается одноосный кристалл с направлением распространения падающего пучка нормально поверхности кристалла и фиксированным значением угла отклонения необыкновенного луча.

1.2 О языке JavaScript

JavaScript — это интерпретируемый язык программирования с объектно ориентированными возможностями. Он был представлен в 1995 году как способ добавлять программы на веб-страницы в браузере Netscape Navigator. С тех пор язык прижился во всех основных графических браузерах. Он дал возможность появиться современным веб-приложениям — браузерные e-mail-клиенты, карты, социальные сети. JavaScript практически не имеет отношения к другому языку под названием Java. Похожее имя было выбрано из маркетинговых соображений [1].

С точки зрения синтаксиса базовый язык JavaScript напоминает C, C++ и Java такими программными конструкциями, как инструкция `if`, цикл `while` и оператор `&&`. Однако это подобие ограничивается синтаксической схожестью. JavaScript — это нетипизированный язык, то есть не требуется определять типы переменных.

JavaScript программа может реагировать на действия пользователя: ввод значения в текстовое поле или щелчок мышью в области изображения в документе. Это достигается путем определения обработчиков событий для документа — фрагментов JavaScript кода, исполняемых при возникновении определенного события, например щелчка на кнопке.

Базовая методика отладки JavaScript кода совпадает с методикой для многих других языков: вставка в код инструкций, которые будут выводить значения нужных переменных так, чтобы можно было понять, что же на самом деле происходит. Иногда для этих целей можно использовать метод `document.write()` или `alert()`. Также удобно пользоваться отладчиками самих браузеров [2].

1.3 Canvas

Canvas - это низкоуровневый интерфейс программирования приложений, работающий в непосредственном режиме. Canvas представляет базовый функционал: например, любой прямоугольник в нём - это всего лишь простая фигура. Тем не менее такой локальный функционал легко дополнить с помощью скриптов JavaScript. Рисовальные команды Canvas выполняются в момент вызова. В Canvas отсутствует непосредственная структура данных с иерархией объектов. Это означает, что графические операции можно выстраивать в неограниченное количество уровней без негативного влияния на производительность приложения. Элемент Canvas поддерживается в большинстве популярных браузеров, таких как Firefox, Chrome, Opera, Safari и Internet Explorer начиная с 9-ой версии [3].

Глава 2

О самой программе

2.1 Возможности

Данная программа является иллюстрацией явления двойного лучепреломления. В графическом окне расположены кристалл, поляризатор и экран. Предусмотрена возможность менять последовательность поляроида и кристалла, поворачивать их вокруг направления падения луча, менять толщину кристалла и при этом наблюдать изменение распределения интенсивностей обыкновенного и необыкновенного лучей.

Вот три возможных случая взаимного расположения объектов (Рис.2.1), (Рис.2.2), (Рис.2.3):

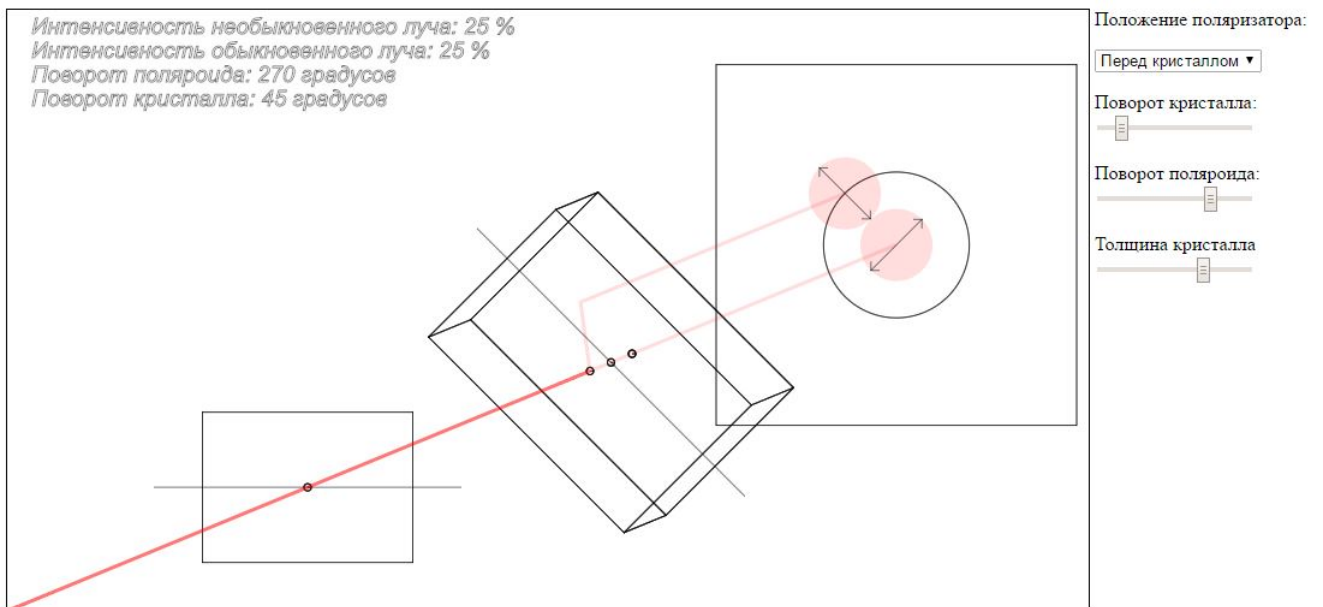


Рис. 2.1: Вид окна программы при кристалле, расположенном за поляридом.

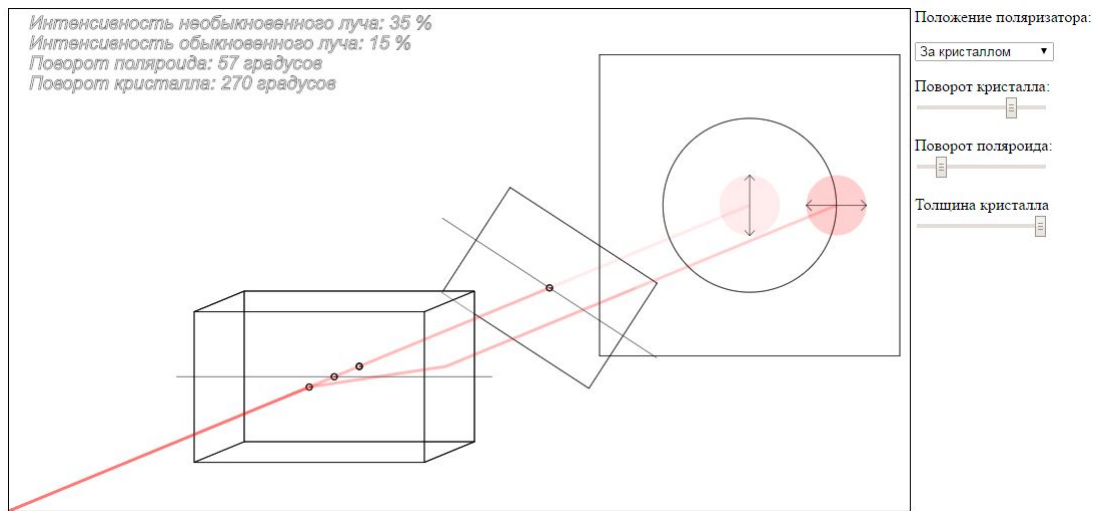


Рис. 2.2: Вид окна программы при кристалле, расположенном перед поляроидом.

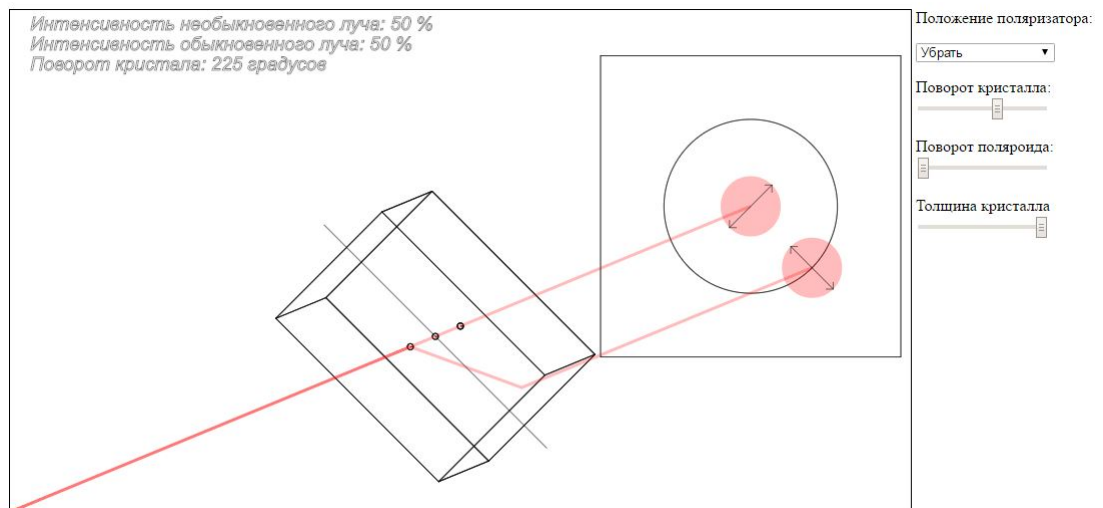


Рис. 2.3: Вид окна программы при отсутствии поляроида.

Работу программы можно проиллюстрировать с помощью следующей блок-схемы (Рис.2.4):

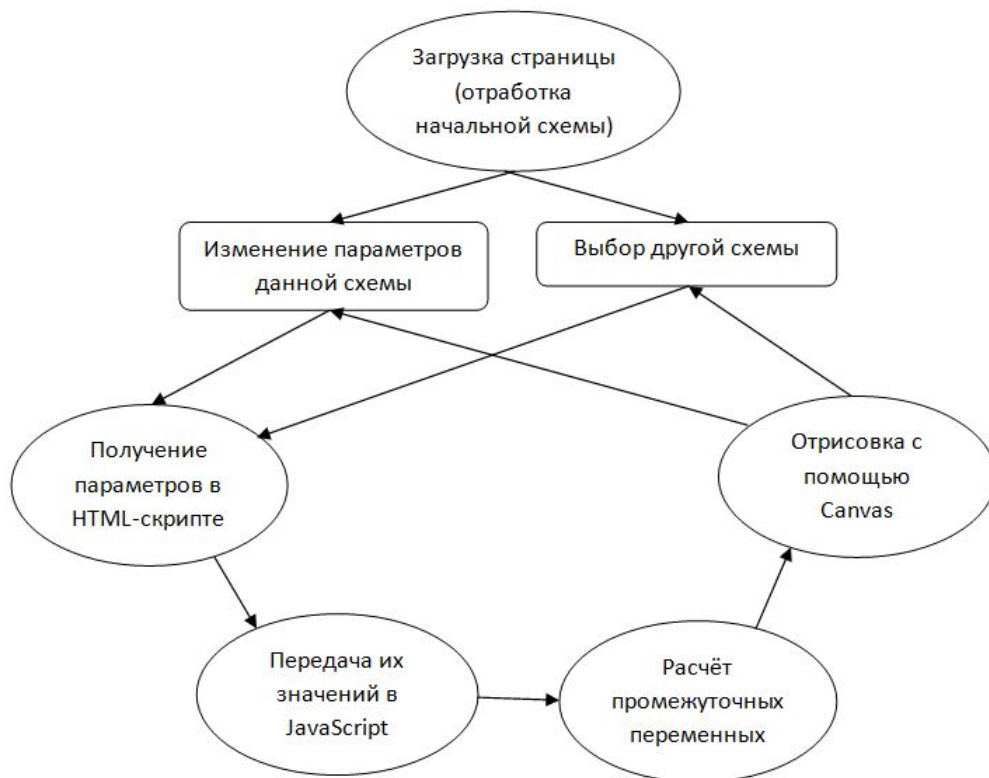


Рис. 2.4: Вид окна программы при отсутствии поляроида.

2.2 Прорисовка объектов

Прорисовка объектов осуществляется с помощью специальных функций, написанных для различных случаев взаимного расположения объектов.

```

//Описываем функцию, рисующую экран:
var screen = function(x0, y0, width, height, thick, alpha){
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
canvas.fillStyle = 'blue';
x1 = x0 - width/2;
y1 = y0 - height/2;
x2 = x0 + width/2;
y2 = y0 - height/2;
x3 = x0 + width/2;
y3 = y0 + height/2;
x4 = x0 - width/2;
y4 = y0 + height/2;
canvas.moveTo(x1, y1);
canvas.lineTo(x2, y2);
canvas.lineTo(x3, y3);

```

```

canvas.lineTo(x4, y4);
canvas.lineTo(x1, y1);
L = thick*Math.tan(alpha*Math.PI/180);
canvas.moveTo(x0+L, y0);
canvas.arc(x0, y0, L, 0, 2*Math.PI);
canvas.stroke();
canvas.stroke();
}

```

```

//Описываем функцию, рисующую анизотропный кристалл:
var parpip = function(x0, y0, width, height, D_fi, thick, alpha) {
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
canvas.strokeStyle="black";
var r = Math.sqrt((width/2.0)*(width/2.0) + (height/2.0)*(height/2.0));
var fi = Math.atan(width/height);
var Dfi = (D_fi*Math.PI)/180.0;
var x01 = x0 - r * Math.sin(fi+Dfi);
var y01 = y0 - r * Math.cos(fi+Dfi);
var x02 = x0 - r * Math.sin(2.0*Math.PI-fi+Dfi);
var y02 = y0 - r * Math.cos(2.0*Math.PI-fi+Dfi);
var x03 = x0 - r * Math.sin(Math.PI+fi+Dfi);
var y03 = y0 - r * Math.cos(Math.PI+fi+Dfi);
var x04 = x0 - r * Math.sin(Math.PI-fi+Dfi);
var y04 = y0 - r * Math.cos(Math.PI-fi+Dfi);
x00 = x0 - thick;
y00 = getY(x00, 500, alpha);
var x11 = x00 - r * Math.sin(fi+Dfi);
var y11 = y00 - r * Math.cos(fi+Dfi);
var x12 = x00 - r * Math.sin(2.0*Math.PI-fi+Dfi);
var y12 = y00 - r * Math.cos(2.0*Math.PI-fi+Dfi);
var x13 = x00 - r * Math.sin(Math.PI+fi+Dfi);
var y13 = y00 - r * Math.cos(Math.PI+fi+Dfi);
var x14 = x00 - r * Math.sin(Math.PI-fi+Dfi);
var y14 = y00 - r * Math.cos(Math.PI-fi+Dfi);
canvas.moveTo(x01, y01);
canvas.lineTo(x02, y02);
canvas.lineTo(x03, y03);
canvas.lineTo(x04, y04);

```

```

canvas.lineTo(x01, y01);
canvas.moveTo(x01, y01);
canvas.lineTo(x11, y11);
canvas.moveTo(x02, y02);
canvas.lineTo(x12, y12);
canvas.moveTo(x03, y03);
canvas.lineTo(x13, y13);
canvas.moveTo(x04, y04);
canvas.lineTo(x14, y14);
canvas.stroke();
canvas.moveTo(x11, y11);
canvas.lineTo(x12, y12);
canvas.lineTo(x13, y13);
canvas.lineTo(x14, y14);
canvas.lineTo(x11, y11);
canvas.globalAlpha = 0.53;
canvas.fillStyle = '#FFF';
canvas.fill();
canvas.stroke();
canvas.moveTo(x0, y0);
canvas.arc(x0, y0, 3, 0, 2*Math.PI);
canvas.moveTo(x00, y00);
canvas.arc(x00, y00, 3, 0, 2*Math.PI);
canvas.moveTo((x0 - (x0-x00))/2, y0 + (y00-y0)/2);
canvas.arc((x0 - (x0-x00))/2, y0 + (y00-y0)/2, 3, 0, 2*Math.PI);
canvas.stroke();
canvas.moveTo(x0, y0);
canvas.arc(x0, y0, 3, 0, 2*Math.PI);
var R = r + 20;
var X1 = x0 - (x0-x00)/2 - R * Math.sin(Dfi);
var Y1 = y0 + (y00-y0)/2 - R * Math.cos(Dfi);
var X2 = x0 - (x0-x00)/2 - R * Math.sin(Math.PI+Dfi);
var Y2 = y0 + (y00-y0)/2 - R * Math.cos(Math.PI+Dfi);
kKr = Math.tan(Math.PI/2 + Dfi);
yKr = y0 + kKr*x0;
canvas.moveTo(X1, Y1);
canvas.lineTo(X2, Y2);
canvas.strokeStyle="black";
canvas.stroke();

```

```
}
```

```
//Описываем функцию, рисующую поляризатор:
```

```
var polarizer = function(x0, y0, width, height, D_fi) {
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
var r = Math.sqrt((width/2.0)*(width/2.0) + (height/2.0)*(height/2.0));
var fi = Math.atan(width/height);
var Dfi = (D_fi*Math.PI)/180.0;
var x1 = x0 - r * Math.sin(fi+Dfi);
var y1 = y0 - r * Math.cos(fi+Dfi);
var x2 = x0 - r * Math.sin(2.0*Math.PI-fi+Dfi);
var y2 = y0 - r * Math.cos(2.0*Math.PI-fi+Dfi);
var x3 = x0 - r * Math.sin(Math.PI+fi+Dfi);
var y3 = y0 - r * Math.cos(Math.PI+fi+Dfi);
var x4 = x0 - r * Math.sin(Math.PI-fi+Dfi);
var y4 = y0 - r * Math.cos(Math.PI-fi+Dfi);
canvas.moveTo(x1, y1);
canvas.lineTo(x2, y2);
canvas.lineTo(x3, y3);
canvas.lineTo(x4, y4);
canvas.lineTo(x1, y1);
canvas.moveTo(x0, y0);
canvas.arc(x0, y0, 3, 0, 2*Math.PI);
canvas.stroke();
var R = r + 20;
var X1 = x0 - R * Math.sin(Dfi);
var Y1 = y0 - R * Math.cos(Dfi);
var X2 = x0 - R * Math.sin(Math.PI+Dfi);
var Y2 = y0 - R * Math.cos(Math.PI+Dfi);
canvas.moveTo(X1, Y1);
canvas.lineTo(X2, Y2);
canvas.stroke();
}
```

```
//Описываем функцию, рисующую лазерный луч:
```

```
var laserLine = function(X0, Y0, alpha, Xe) {
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
```

```

var k = Math.atan(Math.PI*alpha/180);
var Ye = Y0 - k*(Xe-X0);
canvas.strokeStyle="red";
canvas.moveTo(X0, Y0);
canvas.lineTo(Xe, Ye);
canvas.lineWidth = 3;
canvas.stroke();
}

```

//Описываем функцию, рисующую обыкновенный луч:

```

var OBeam = function(X0, Y0, alpha, Xe, id) {
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
var k = Math.atan(Math.PI*alpha/180);
var Ye = Y0 - k*(Xe-X0);
if(id == 'Before') {
Io = 1 - Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0)*
Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0);
canvas.strokeStyle='rgba(255, 0, 0, ' + 0.5*Io + ')';
}
if(id == 'After') Io = 0.5*(1 - Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0)*
Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0));
canvas.strokeStyle='rgba(255, 0, 0, ' + Io + ')';
}
if(id == 'Behind') {
canvas.strokeStyle='rgba(255, 0, 0, 0.5)';
}
if(id == 'Without') canvas.strokeStyle='rgba(255, 0, 0, 0.5)';
}
canvas.moveTo(X0, Y0);
canvas.lineTo(Xe, Ye);
canvas.lineWidth = 3;
canvas.stroke();
}

```

//Описываем функцию, рисующую условное изображение обыкновенного луча на экране:

```

var OCycle = function(X0, Y0, alpha, Xe, id) {
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();

```

```

var k = Math.atan(Math.PI*alpha/180);
var Ye = Y0 - k*(Xe-X0);
if(id == 'Before') {
Io = 1 - Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)*
Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)
canvas.fillStyle='rgba(255, 0, 0, ' + 0.5*Io + ')';
}
if(id == 'Behind') {
Io = 0.5*(1 - Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0)*
Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0));
canvas.fillStyle='rgba(255, 0, 0, ' + Io + ')';
}
if(id == 'Without') {
canvas.fillStyle='rgba(255, 0, 0, 0.5)';
}
canvas.moveTo(Xe+30, Ye);
canvas.arc(Xe, Ye, 30, 0, 2*Math.PI);
canvas.fill();
}

```

//Описываем функцию, рисующую на экране стрелки для направления поляризации
обыкновенного луча:

```

var OArrow = function(X0, Y0, alpha, Xe, Dfi)
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
var k = Math.atan(Math.PI*alpha/180.0);
var Ye = Y0 - k*(Xe-X0);
var X1 = Xe - 30 * Math.sin(Dfi*Math.PI/180 - Math.PI/2);
var Y1 = Ye - 30 * Math.cos(Dfi*Math.PI/180 - Math.PI/2);
var X2 = Xe + 30.0 * Math.sin(Dfi*Math.PI/180 - Math.PI/2);
var Y2 = Ye + 30.0 * Math.cos(Dfi*Math.PI/180 - Math.PI/2);
canvas.moveTo(X1, Y1);
canvas.lineTo(X2, Y2);
var X1_Arw1 = Xe - 30 * Math.sin(Dfi*Math.PI/180 - Math.PI/2) +
7 * Math.sin(Dfi*Math.PI/180 - Math.PI/4);
var Y1_Arw1 = Ye - 30 * Math.cos(Dfi*Math.PI/180 - Math.PI/2) +
7 * Math.cos(Dfi*Math.PI/180 - Math.PI/4);
var X1_Arw2 = Xe - 30 * Math.sin(Dfi*Math.PI/180 - Math.PI/2) -
7 * Math.sin(Dfi*Math.PI/180 + Math.PI/4);

```

```

var Y1_Arw2 = Ye - 30 * Math.cos(Dfi*Math.PI/180 - Math.PI/2) -
7 * Math.cos(Dfi*Math.PI/180 + Math.PI/4);
canvas.moveTo(X1, Y1);
canvas.lineTo(X1_Arw1, Y1_Arw1);
canvas.moveTo(X1, Y1);
canvas.lineTo(X1_Arw2, Y1_Arw2);
var X2_Arw1 = Xe + 30 * Math.sin(Dfi*Math.PI/180 - Math.PI/2) +
7 * Math.sin(Dfi*Math.PI/180 + Math.PI/4);
var Y2_Arw1 = Ye + 30 * Math.cos(Dfi*Math.PI/180 - Math.PI/2) +
7 * Math.cos(Dfi*Math.PI/180 + Math.PI/4);
var X2_Arw2 = Xe + 30 * Math.sin(Dfi*Math.PI/180 - Math.PI/2) -
7 * Math.sin(Dfi*Math.PI/180 - Math.PI/4);
var Y2_Arw2 = Ye + 30 * Math.cos(Dfi*Math.PI/180 - Math.PI/2) -
7 * Math.cos(Dfi*Math.PI/180 - Math.PI/4);
canvas.moveTo(X2, Y2);
canvas.lineTo(X2_Arw1, Y2_Arw1);
canvas.moveTo(X2, Y2);
canvas.lineTo(X2_Arw2, Y2_Arw2);
canvas.strokeStyle="black";
canvas.lineWidth = 1;
canvas.stroke();
}

//Описываем функцию, рисующую вращающийся необыкновенный луч:
var EBeam = function(X0, Y0, alpha, thick, Dfi, k, XX, id) {
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
L = thick*Math.tan(alpha*Math.PI/180);
dX = -L*Math.sin(Dfi*Math.PI/180.0);
dY = L*Math.cos(Dfi*Math.PI/180.0);
var Xe = X0 + thick + dX;
var Ye = getY(X0+thick, 500.0, k) - dY;
var YY = getY(XX, 500.0, k);
YYY = YY;
XXX = XX;
if(id == 'Before') {
Ie = Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)*
Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0);
canvas.strokeStyle='rgba(255, 0, 0, ' + 0.5*Ie + ')';

```

```

canvas.moveTo(X0, Y0);
canvas.lineTo(Xe, Ye);
canvas.lineTo(XX+dX,YY-dY);
}
if(id == 'After') {
Ie = 0.5*Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0)*
Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0);
canvas.strokeStyle='rgba(255, 0, 0, ' + Ie + ')';
canvas.moveTo(Xe-thick, Y0-dY);
canvas.lineTo(XX+dX,YY-dY);
}
if(id == 'Behind') {
canvas.strokeStyle='rgba(255, 0, 0, 0.5)';
canvas.moveTo(X0, Y0);
canvas.lineTo(Xe, Ye);
canvas.lineTo(XX+dX,YY-dY);
}
if(id == 'Without') {
canvas.strokeStyle='rgba(255, 0, 0, 0.5)';
canvas.moveTo(X0, Y0);
canvas.lineTo(Xe, Ye);
canvas.lineTo(XX+dX,YY-dY);
}
canvas.lineWidth = 3;
canvas.stroke();
}

```

//Описываем функцию, рисующую условное изображение необыкновенного луча на экране:

```

var ECycle = function(id) {
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
if(id == 'Before') {
Ie = Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)*
Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0);
canvas.fillStyle='rgba(255, 0, 0, ' + 0.5*Ie + ')';
}
if(id == 'Behind') {
Ie = 0.5*Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)*

```



```

Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0);
canvas.fillStyle='rgba(255, 0, 0, ' + Ie + ')';
}
if(id == 'Without') {
canvas.fillStyle='rgba(255, 0, 0, 0.5)';
}
canvas.moveTo(XXX+dX+30,YYY-dY);
canvas.arc(XXX+dX, YYY-dY, 30, 0, 2*Math.PI);
canvas.fill();
}

```

//Описываем функцию, рисующую на экране стрелки для направления поляризации
необыкновенного луча:

```

var EArrow = function(Dfi) {
var canvas = document.getElementById("draw").getContext('2d');
canvas.beginPath();
var X1 = XXX+dX - 30 * Math.sin(Dfi*Math.PI/180);
var Y1 = YYY-dY - 30 * Math.cos(Dfi*Math.PI/180);
var X2 = XXX+dX + 30.0 * Math.sin(Dfi*Math.PI/180);
var Y2 = YYY-dY + 30.0 * Math.cos(Dfi*Math.PI/180);
canvas.moveTo(X1, Y1);
canvas.lineTo(X2, Y2);
var X1_Arw1 = XXX+dX - 30 * Math.sin(Dfi*Math.PI/180) -
7 * Math.sin(-Dfi*Math.PI/180 - Math.PI/4);
var Y1_Arw1 = YYY-dY - 30 * Math.cos(Dfi*Math.PI/180) +
7 * Math.cos(-Dfi*Math.PI/180 - Math.PI/4);
var X1_Arw2 = XXX+dX - 30 * Math.sin(Dfi*Math.PI/180) -
7 * Math.sin(-Dfi*Math.PI/180 + Math.PI/4);
var Y1_Arw2 = YYY-dY - 30 * Math.cos(Dfi*Math.PI/180) +
7 * Math.cos(-Dfi*Math.PI/180 + Math.PI/4);
canvas.moveTo(X1, Y1);
canvas.lineTo(X1_Arw1, Y1_Arw1);
canvas.moveTo(X1, Y1);
canvas.lineTo(X1_Arw2, Y1_Arw2);
var X2_Arw1 = XXX+dX + 30 * Math.sin(Dfi*Math.PI/180) +
7 * Math.sin(-Dfi*Math.PI/180 + Math.PI/4);
var Y2_Arw1 = YYY-dY + 30 * Math.cos(Dfi*Math.PI/180) -
7 * Math.cos(-Dfi*Math.PI/180 + Math.PI/4);
var X2_Arw2 = XXX+dX + 30 * Math.sin(Dfi*Math.PI/180) +

```

```

7 * Math.sin(-Dfi*Math.PI/180 - Math.PI/4);
var Y2_Arw2 = YYY-dY + 30 * Math.cos(Dfi*Math.PI/180) -
7 * Math.cos(-Dfi*Math.PI/180 - Math.PI/4);
canvas.moveTo(X2, Y2);
canvas.lineTo(X2_Arw1, Y2_Arw1);
canvas.moveTo(X2, Y2);
canvas.lineTo(X2_Arw2, Y2_Arw2);
canvas.strokeStyle="black";
canvas.lineWidth = 1;
canvas.stroke();
}

```

В программе также используются отдельные функции, осуществляющие необходимые вычисления и преобразования переменных:

```

var getAngle = function(D_fi_1, D_fi_2) {
dFi_Cr = D_fi_1;
dFi_Pol = D_fi_2;
}

//Описываем функцию, определяющую Y-координату на прямой:
var getY = function(posX, posY, angle) {
var k = Math.atan(Math.PI*angle/180.0);
return posY - k*posX;
}

//Объявлены глобальные переменные:
var kKr, yKr;
var dX, dY;
var XXX, YYY;
var L = 0;
var Ie = 0;
var Io = 0;
var dFi_Cr, dFi_Pol;

```

2.3 Вставка JavaScript в код HTML

В отдельном файле описана разметка страницы, холст canvas и вставлен код JavaScript с помощью тега `<script>`. В нём с разными параметрами вызваны функции, описывающие графические объекты. С помощью тега `<select>` осуществляется выбор между раз-

личными способами расположения объектов на холсте.

```

<html>
<head>
<meta charset="UTF-8»
<script src="functions.js»</script>
</head>
<body>
<table>
<td valign="top»
<canvas id="draw"width="900"height="500"style="border:1px solid»</canvas>
</td>
<td valign="top»
<p size="7»Положение поляризатора:</p>
<p><select idname='Value' id='myselect' onchange="fc(document.getElementById('myselect')).
<option selected value='1'>Перед кристаллом</option>
<option value='2'>За кристаллом</option>
<option value='3'>Убрать</option>
</select></p>
<p size="4»
Поворот кристалла: <br><input type='range' id='Dfi_Cr' min='0' max='360'
step='1' value='0' oninput = "fc(document.getElementById('myselect').value);»
</p>
<p size="4»Поворот поляроида: <br><input type='range' id='Dfi_Pol' min='0' max='360'
step='1' value='0' oninput = "fc(document.getElementById('myselect').value);»</p>
<p>Толщина кристалла<br><input type='range' id='Thick' min='0.0' max='50.0'
step='1.0' value='35.0' oninput = "fc(document.getElementById('myselect').value);»</p>
</td>
</table>
</body>
<script type="text/javascript»
fc(document.getElementById('myselect').value);
function fc(val) {
var dFi_Cr = document.getElementById("Dfi_Cr").value;
var dFi_Pol = document.getElementById("Dfi_Pol").value;
var thick = Math.round(document.getElementById("Thick").value);
getAngle(dFi_Cr, dFi_Pol);
var canvas = document.getElementById("draw").getContext('2d');
canvas.clearRect(0, 0, 900, 500);

```

```

canvas.strokeStyle="black";
canvas.lineWidth=1;
canvas.font = "italic 14pt Arial";
if (val == '1') {
var Draw1 = function() {
var p1=250.0, p2=600.0, p3 = 740.0, p4 = 520.0, k = 25.0, alpha = 60.0, id = 'Before';
polarizer(p1, getY(p1, 500, k), 125, 175, dFi_Pol);
parpip(p4, getY(p4, 500, k), 150, 230, dFi_Cr, thIck, k);
screen(p3, getY(p3, 500, k), 300, 300, thIck, alpha);
laserLine(0,500, k, p4-thIck);
OBeam(p4-thIck, getY(p4-thIck, 500, k), k, p3, id);
OCycle(p4-thIck, getY(p4-thIck, 500, k), k, p3, id);
OArrow(p4-thIck, getY(p4-thIck, 500, k), k, p3, dFi_Cr);
EBeam(p4-thIck, getY(p4-thIck, 500.0, k), alpha, thIck, dFi_Cr, k, p3, id);
ECycle(id);
EArrow(dFi_Cr);
canvas.strokeText("Интенсивность необыкновенного луча: "+
Math.round(0.5*Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)*
Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)*100) + "
canvas.strokeText("Интенсивность обыкновенного луча: "+
Math.round(0.5*(1 - Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0)*
Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0))*100) + "
canvas.strokeText("Поворот поляроида: "+ dFi_Pol + "градусов 20, 60);
canvas.strokeText("Поворот кристалла: "+ dFi_Cr + "градусов 20, 80);
}
Draw1();
}
if(val == '2')
var Draw2 = function() {
var p1=350.0, p3 = 740.0, p4 = 540.0, k = 25.0, alpha = 60.0, id = 'Behind';
polarizer(p4, getY(p4, 500, k), 125, 175, dFi_Pol);
parpip(p1, getY(p1, 500, k), 150, 230, dFi_Cr, thIck, k);
screen(p3, getY(p3, 500, k), 300, 300, thIck, alpha);
laserLine(0,500, k, p1-thIck);
OBeam(p1-thIck, getY(p1-thIck, 500, k), k, p4, id);
OBeam(p4, getY(p4, 500, k), k, p3, 'After');
OCycle(p1-thIck, getY(p1-thIck, 500, k), k, p3, id);
OArrow(p1-thIck, getY(p1-thIck, 500, k), k, p3, dFi_Cr);
EBeam(p1-thIck, getY(p1-thIck, 500.0, k), alpha, thIck, dFi_Cr, k, p4, id);
}
}

```

```

EBeam(p4, getY(p4, 500.0, k), alpha, thIck, dFi_Cr, k, p3, 'After');
ECycle(id);
EArrow(dFi_Cr);
canvas.strokeText("Интенсивность необыкновенного луча: "+
Math.round(0.5*Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)*
Math.cos((dFi_Pol-dFi_Cr)*Math.PI/180.0)*100) + "
canvas.strokeText("Интенсивность обыкновенного луча: "+
Math.round(0.5*(1 - Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0)*
Math.cos((dFi_Cr-dFi_Pol)*Math.PI/180.0))*100) + "
canvas.strokeText("Поворот поляроида: "+ dFi_Pol + "градусов 20, 60);
canvas.strokeText("Поворот кристалла: "+ dFi_Cr + "градусов 20, 80);
}
Draw2();
}
if(val == '3') {
var Draw3 = function() {
var p3 = 740.0, p4 = 450.0, k = 25.0, alpha = 60.0, id = 'Without';
parpip(p4, getY(p4, 500, k), 150, 230, dFi_Cr, thIck, k);
screen(p3, getY(p3, 500, k), 300, 300, thIck, alpha);
laserLine(0,500, k, p4-thIck);
OBeam(p4-thIck, getY(p4-thIck, 500, k), k, p3, id);
OCycle(p4-thIck, getY(p4-thIck, 500, k), k, p3, id);
OArrow(p4-thIck, getY(p4-thIck, 500, k), k, p3, dFi_Cr);
EBeam(p4-thIck, getY(p4-thIck, 500.0, k), alpha, thIck, dFi_Cr, k, p3, id);
ECycle(id);
EArrow(dFi_Cr);
canvas.strokeText("Интенсивность необыкновенного луча: "+ 50 + "
canvas.strokeText("Интенсивность обыкновенного луча: "+ 50 + "
canvas.strokeText("Поворот кристалла: "+ dFi_Cr + "градусов 20, 60);
}
Draw3();
}
}
fc(document.getElementById('myselect').value);
</script>
</div>
</body>
</html>

```

Итоги работы

В результате выполнения работы были использованы полученные в курсе "Основы Web-технологий" знания и навыки, проведено более глубокое ознакомление с методами визуализации с помощью языка JavaScript и разработана программы для наглядной визуализации данного явления и дальнейшего использования в преподавательских целях. В качестве недостатков программы можно отметить отсутствие дизайна и стилизованного интерфейса. Перспективами дальнейшей работы являются как раз устранение этих недостатков, возможность задавать тип кристалла, а соответственно, угол отклонения необыкновенного луча и осуществление возможности пользователя расставлять объекты самостоятельно, а также вращать их с помощью курсора мыши.

Литература

- [1] Выразительный Javascript, 2-е издание, Marijn Haverbeke (2015), No Starch Press ISBN: 978-1-59327-584-6.
- [2] Флэнаган Д., JavaScript. Подробное руководство. — Пер. с англ. — СПб: Символ Плюс, 2008. — 992 с., ил., ISBN 10: 5 93286 103 7, ISBN 13: 978 5 93286 103 5.
- [3] Рафаэлло Чекко Ч-37, Графика на JavaScript. — СПб.: Питер, 2013. — 272 с.: ил. ISBN 978-5-4461-0034-7.
- [4] Иродов И. Е., Волновые процессы. Основные законы — 4-е издание — М.: БИНОМ. Лаборатория знаний 2007. — 263 с.: ил. ISBN 978-5-94774-692-1.