

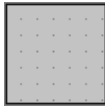
Задание для группы ЭН-35

На этой лабораторной работе вашей задачей будет демонстрация умения рисовать с применением GDI WinAPI. Задания основываются на предыдущей лабораторной работе с некоторыми дополнениями. Весь необходимый новый материал излагается ниже, после заданий.

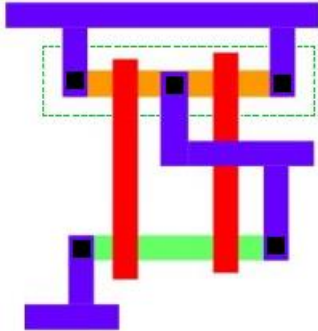
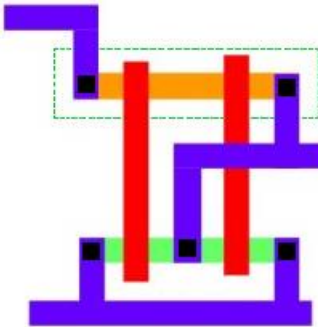
ЗАДАНИЯ НА МИНИМАЛЬНЫЙ БАЛЛ (7 баллов)

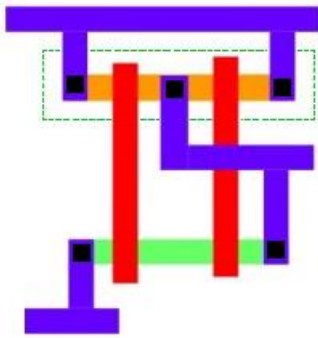
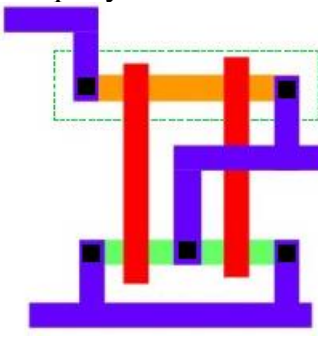
В процессе выполнения лабораторной работы на 7 баллов необходимо получить три плюса, выполнив следующие этапы:

1. **Нарисовать рабочую область**, в которой будет отображаться топология. Рабочая область рисуется серым цветом, на ней пикселями более тёмного цвета рисуется сетка, расстояние между точками – 20 пикселей.



2. **Нарисовать топологию согласно варианту.**

Вариант	Задание
1, 5, 9, 13, 17, 21, 25	<p>Вы рисуете топологию элемента NAND2:</p>  <p>Элемент можно перемещать кликом правой кнопки мышки с зажатой клавишей Shift. Элемент нельзя переместить, если он выходит за пределы окна.</p>
2, 6, 10, 14, 18, 22, 26	<p>Вы рисуете топологию элемента NOR2:</p>  <p>Элемент можно перемещать кликом правой кнопки мышки с зажатой клавишей Ctrl. Элемент нельзя переместить, если он выходит за пределы окна.</p>

<p>3, 7, 11, 15, 19, 23, 27</p>	<p>Вы рисуете топологию элемента NAND2:</p>  <p>Элемент можно перемещать кликом левой кнопки мышки с зажатой клавишей Shift. Элемент нельзя переместить, если он выходит за пределы окна.</p>
<p>4, 8, 12, 16, 20, 24, 28</p>	<p>Вы рисуете топологию элемента NOR2:</p>  <p>Элемент можно перемещать кликом левой кнопки мышки с зажатой клавишей Ctrl. Элемент нельзя переместить, если он выходит за пределы окна.</p>

3. Нарисовать график согласно варианту.

Вариант	Задание
<p>1, 5, 9, 13, 17, 21, 25</p>	<p>В правой нижней четверти окна рисуется график косинуса. Число периодов равно 4. Под графиком рисуется область (график рисуется не прямо поверх сетки). Цвета – по выбору разработчика. При клике в область графика должен появляться диалог с информацией о том, что по графику кликать нельзя.</p>
<p>2, 6, 10, 14, 18, 22, 26</p>	<p>В правой верхней четверти окна рисуется график синуса. Число периодов равно 5. Под графиком рисуется область (график рисуется не прямо поверх сетки). Цвета – по выбору разработчика. При клике в область графика должен появляться диалог с информацией о том, что по графику кликать нельзя.</p>
<p>3, 7, 11, 15, 19, 23, 27</p>	<p>В левой верхней четверти окна рисуется график синуса. Число периодов равно 5. Под графиком рисуется область (график рисуется не прямо поверх сетки). Цвета – по выбору разработчика. При клике в область графика должен появляться диалог с информацией о том, что по графику кликать нельзя.</p>
<p>4, 8, 12,</p>	<p>В левой нижней четверти окна рисуется график косинуса. Число</p>

16, 20, 24, 28	периодов равно 4. Под графиком рисуется область (график рисуется не прямо поверх сетки). Цвета – по выбору разработчика. При клике в область графика должен появляться диалог с информацией о том, что по графику кликать нельзя.
-------------------	--

ЗАДАНИЯ НА ДОПОЛНИТЕЛЬНЫЕ БАЛЛЫ (+3 балла максимум)

Каждый из пунктов, чтобы не запутаться в коде, реализуется в виде отдельной функции, которые должны вызываться из обработчика WM_PAINT. Имеется в виду, что в WM_PAINT не должно быть ни строчки кода, ответственного за отрисовку, только код типа (вы дополняете код всем, что вам необходимо):

```
case WM_PAINT:  
    hdc = BeginPaint(hWnd, &ps);  
  
    DrawGrid(hdc);           // в этой функции рисуется сетка  
    DrawLayout(hdc);        // в этой функции рисуются элементы  
    DrawPlot(hdc);         // в этой функции рисуется график  
  
    EndPaint(hWnd, &ps);  
    break;
```

Необходимо выполнить все задания на минимальное количество баллов. После этого можно выполнить одно или несколько приведённых ниже дополнительных заданий.

- **Плюс балл.** Клавишами «+» и «-» должно регулироваться число периодов графика. Минимум – 1, максимум - 6.
- **Плюс балл.** Перемещение, задаваемое номером варианта, может выполняться только в случае, если топология «выбрана». Для выбора топологии сначала необходимо просто кликнуть в область ограничивающего прямоугольника топологии – вокруг неё рисуется белая рамка. Клик вне топологии – выделение снимается.
- **Плюс балл.** Информация о положении элемента на окне должна сохраняться при закрытии окна и считываться при его открытии из конфигурационного файла. Формат – на выбор разработчика.

ЗАДАНИЕ ПОВЫШЕННОЙ СЛОЖНОСТИ

Кликав в область топологии, я должен иметь возможность выбрать не всю топологию (не топологию всего вентиля целиком), а отдельные полигоны, из которых строится топология. При этом выбранный полигон рисуется своим цветом, а все остальные (не выбранные) – градациями серого согласно цвету.

Клик вне полигонов – возврат к обычному способу рисования.

Дополнительные сведения для выполнения лабораторной работы.

Q: Как получить размер клиентской области окна (понадобится для отрисовки сетки)?

A: Получить размер клиентской области окна можно с помощью вызова функции `GetClientRect`, в которую нужно передать дескриптор окна, размер клиентской области которого вычисляется, и ссылку на структуру `RECT`.

Пример того, как пользоваться этой функцией, и результат работы приведён ниже (необходимые строки выделены жирным шрифтом). В этом примере определяется размер клиентской области окна и рисуется прямоугольник в размер окна с отступами по 10 пикселей.

```
LRESULT CALLBACK WndProc(HWND hWnd, UINT message, WPARAM wParam, LPARAM lParam) {
    PAINTSTRUCT ps;
    HDC hdc;

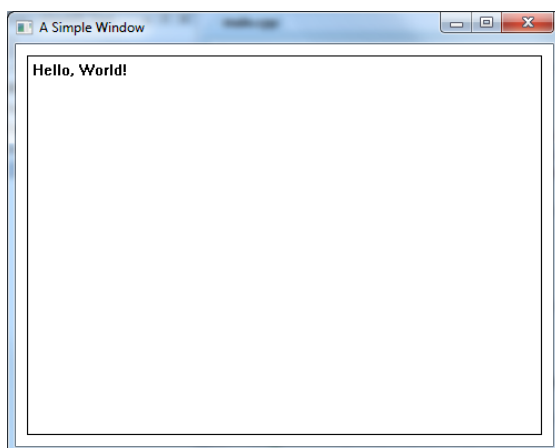
    RECT rect;
    GetClientRect(hWnd, &rect);

    switch (message) {
    case WM_PAINT:
        hdc = BeginPaint(hWnd, &ps);

        Rectangle(hdc, rect.left + 10, rect.top + 10, rect.right - 10, rect.bottom - 10);

        TextOut(hdc, 15, 15, stringToDraw, strlen(stringToDraw));

        EndPaint(hWnd, &ps);
        break;
    case WM_DESTROY:
        PostQuitMessage(0);
        break;
    default:
        return DefWindowProc(hWnd, message, wParam, lParam);
    }
    return 0;
}
```



Q: Как определить, что при клике мышкой зажата клавиша `Shift`? (`Ctrl` – по аналогии).

A: Нужно посмотреть, что приходит в переменной `wParam`.

Пример.

```
case WM_LBUTTONDOWN:
    if (wParam & MK_SHIFT) {
        // Тут пишем код, который должен выполняться, если в момент клика нажата клавиша Shift
    }
    break;
```