

Politechnika Wrocławska  
Wydział Elektroniki  
Urządzenia Peryferyjne

---

## OBSŁUGA SKANERA PŁASKIEGO (WIA 2.0)

---

*Termin zajęć:*

WTOREK TN 7:30

*Autorzy:*

JAKUB CHMIEL 235028

TOMASZ CIEŚLAR 235652

*Prowadzący:*

dr inż. Tomasz Walkowiak

29 listopada 2018

# Spis treści

1	Cel ćwiczenia . . . . .	2
2	Wstęp . . . . .	2
3	Założenia projektowe . . . . .	2
4	Wykorzystane narzędzia . . . . .	2
5	Implementacja programu . . . . .	3
	5.1 Interfejs aplikacji . . . . .	3
	5.2 Kod źródłowy . . . . .	4
6	Wnioski . . . . .	9
7	Bibliografia . . . . .	9

# 1 Cel ćwiczenia

1. Sprawdzić czy skaner działa poprawnie
2. Napisać program wykonujący skanowanie przy pomocy skanera płaskiego.  
Możliwości programu obejmują:
  - skanowanie z wykorzystaniem UI
  - skanowanie bez wykorzystania UI
  - wyświetlanie uzyskanego obrazu
  - zmiana trybu skanowania (1-bitowy, skala szarości, RGB)
3. Rozszerzyć działanie programu:
  - zapis skanowanych obrazów do plików graficznych

# 2 Wstęp

Windows Image Acquisition (WIA) jest platformą do przechwytywania obrazów w systemach operacyjnych z rodziny Windows, począwszy od Windowsa Millenium Edition i Windowsa XP.

Platforma WIA pozwala na interakcję między aplikacjami a sprzętem do przechwytywania obrazów i standaryzuje interakcje między różnymi aplikacjami a skanerami. Pozwala to różnym aplikacjom na porozumiewanie się z różnymi skanerami bez konieczności modyfikowania aplikacji przez programistów i sterowników przez producentów skanerów dla każdej kombinacji sprzętu i programów.

# 3 Założenia projektowe

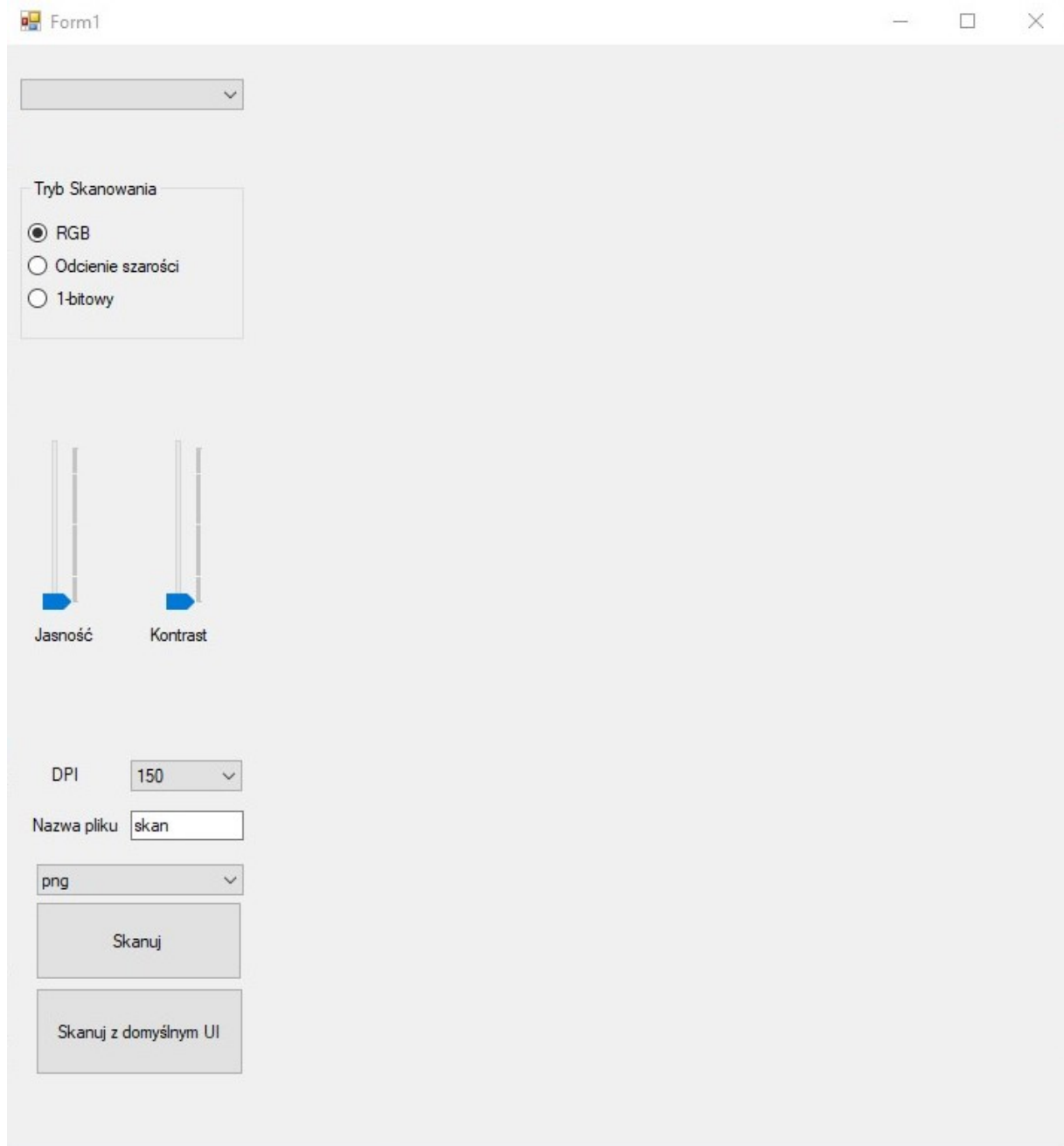
- Program był pisany w języku C#.
- Na komputerze, na którym uruchamiany był program zainstalowano system operacyjny Windows 10 w wersji 64-bitowej.

# 4 Wykorzystane narzędzia

- Windows Forms - API do implementacji interfejsu graficznego dla platformy .NET.
- WIA 2.0 - aktualna wersja platformy WIA, po raz pierwszy wprowadzona w systemie Windows Vista.

## 5 Implementacja programu

### 5.1 Interfejs aplikacji



*Rysunek 1. Interfejs aplikacji*

## 5.2 Kod źródłowy

### Okno aplikacji

```
using [...]  
  
namespace UP_Skaner  
{  
    public partial class Form1 : Form  
    {  
        private ScannerClass device;  
  
        public Form1()  
        {  
            InitializeComponent();  
            ListDevices();  
            device = (ScannerClass)comboBox2.SelectedItem;  
        }  
  
        //wylistowanie wszystkich urzadzen  
        public void ListDevices()  
        {  
            var deviceManager = new DeviceManager();  
  
            for (int i = 1; i <=  
                deviceManager.DeviceInfos.Count; i++)  
            {  
                //zebranie wszystkich urzadzen typu skaner  
                if (deviceManager.DeviceInfos[i].Type ==  
                    WiaDeviceType.ScannerDeviceType)  
                {  
                    comboBox2.Items.Add(new  
                        ScannerClass(deviceManager.  
                            DeviceInfos[i]));  
                }  
            }  
        }  
  
        private void comboBox2_SelectedIndexChanged(object  
            sender, EventArgs e)  
        {  
            device = (ScannerClass)comboBox2.SelectedItem;  
        }  
  
        private void button1_Click(object sender, EventArgs e)  
        {  
            //wyczyszczenie podgladu  
            pictureBox1.Image = null;  
            if (device != null)  
            {  
                var connectedDevice =
```

```

        device._deviceInfo.Connect();
var scannerItem = connectedDevice.Items[1];

ImageFile image;
//ustawienie trybu koloru
if (radioButton1.Checked)
    SetColorMode(scannerItem, 1);
else if(radioButton2.Checked)
{
    SetColorMode(scannerItem, 2);
}
else if (radioButton3.Checked)
{
    SetColorMode(scannerItem, 4);
}

//ustawienie DPI
int dpi = Int32.Parse(comboBox3.Text);
SetDPI(scannerItem, dpi);

//ustawienie rozmiaru
SetHeightWidth(scannerItem,1250,1700);

//ustawienie jasnosci
SetBrightness(scannerItem, trackBar1.Value);

//ustawienie kontrastu
SetContrast(scannerItem, trackBar2.Value);

//rozszerzenia zapisanego pliku
switch (comboBox1.SelectedItem)
{
    case "png":
        image = (ImageFile)
            scannerItem.Transfer
            (FormatID.wiaFormatPNG);
        break;
    case "jpeg":
        image = (ImageFile)scannerItem.Transfer
            (FormatID.wiaFormatJPEG);
        break;
    case "tiff":
        image = (ImageFile)scannerItem.Transfer
            (FormatID.wiaFormatTIFF);
        break;
    default:
        image = (ImageFile)scannerItem.Transfer
            (FormatID.wiaFormatJPEG);
        break;
}

```

```

        //ustawienie nazwy pliku
        var path =
            $"{textBox1.Text}.{comboBox1.SelectedItem}";

        //usuniecie pliku jesli istnieje
        if (File.Exists(path))
        {
            File.Delete(path);
        }

        //zapisanie pliku
        image.SaveFile(path);

        //ustawienie obrazu jako podglad
        pictureBox1.Image = new Bitmap(path);
        pictureBox1.SizeMode =
            PictureBoxSizeMode.StretchImage;
    }
    else
    {
        MessageBox.Show("Wybierz urządzenie");
    }
}

//stale biblioteki WIA
const string WIA_SCAN_COLOR_MODE = "6146";
const string WIA_HORIZONTAL_SCAN_RESOLUTION_DPI =
    "6147";
const string WIA_VERTICAL_SCAN_RESOLUTION_DPI = "6148";

const string WIA_HORIZONTAL_SCAN_START_PIXEL = "6149";
const string WIA_VERTICAL_SCAN_START_PIXEL = "6150";

const string WIA_HORIZONTAL_SCAN_SIZE_PIXELS = "6151";
const string WIA_VERTICAL_SCAN_SIZE_PIXELS = "6152";

const string WIA_SCAN_BRIGHTNESS_PERCENTS = "6154";
const string WIA_SCAN_CONTRAST_PERCENTS = "6155";

//ustawienie trybu kolorow
public void SetColorMode(IItem scannerItem, int mode)
{
    SetWIAProperty(scannerItem.Properties,
        WIA_SCAN_COLOR_MODE, mode );
}

```

```

//ustawienie rozmiarow obrazu
public void SetHeightWidth(IItem scannerItem, int
    width, int height)
{
    SetWIAProperty(scannerItem.Properties,
        WIA_HORIZONTAL_SCAN_SIZE_PIXELS, width);
    SetWIAProperty(scannerItem.Properties,
        WIA_VERTICAL_SCAN_SIZE_PIXELS, height);
}

//ustawienie DPI
public void SetDPI(IItem scannerItem, int dpi)
{
    SetWIAProperty(scannerItem.Properties,
        WIA_HORIZONTAL_SCAN_RESOLUTION_DPI, dpi);
    SetWIAProperty(scannerItem.Properties,
        WIA_VERTICAL_SCAN_RESOLUTION_DPI, dpi);
}

//ustawienie parametru w bibliotece WIA
private static void SetWIAProperty(IProperties props,
    object propName, object propValue)
{
    Property property = props.get_Item(ref propName);
    property.set_Value(ref propValue);
}

//ustawienie jasnosci
public void SetBrightness(IItem scannerItem, int value)
{
    SetWIAProperty(scannerItem.Properties,
        WIA_SCAN_BRIGHTNESS_PERCENTS, value);
}

//ustawienie kontrastu
public void SetContrast(IItem scannerItem, int value)
{
    SetWIAProperty(scannerItem.Properties,
        WIA_SCAN_CONTRAST_PERCENTS, value);
}

//skanowanie z domyslnym UI
private void button2_Click(object sender, EventArgs e)
{
    //format JPEG
    const string wiaFormatJPEG =
        "{B96B3CAE-0728-11D3-9D7B-0000F81EF32E}";
    //nazwa pliku
    String fileName = "skan_bez_UI";

```



```

        //CommonDialog - domyslne okno skanowania systemu
        Windows
        ICommonDialog wiaDiag = new WIA.CommonDialog();
        ImageFile wiaImage = null;
        //pobranie obrazu z okreslonymi ustawieniami
        wiaImage = wiaDiag.ShowAcquireImage(WiaDeviceType.
            UnspecifiedDeviceType,
            WiaImageIntent.GrayscaleIntent,
            WiaImageBias.MaximizeQuality,
            wiaFormatJPEG, true, true, false);
        Vector vector = wiaImage.FileData;
        //ustawienie podgladu
        pictureBox1.Image = Image.FromStream(new
            MemoryStream((byte[])vector.get_BinaryData()));
        //stworzenie obrazu
        Image img = Image.FromStream(new
            MemoryStream((byte[])vector.get_BinaryData()));
        zapis obrazu
        img.Save(fileName + ".JPEG");
    }
}

```

#### Klasa ScannerClass

```

using [...]

namespace scanner
{
    //Klasa pomocnicza. Jej funkcja jest poprawne wypelnienie
    //ComboBoxa do wyboru skanera
    class ScannerClass
    {
        public DeviceInfo _deviceInfo;
        public ScannerClass(DeviceInfo info)
        {
            _deviceInfo = info;
        }

        public override string ToString()
        {
            return (string)_deviceInfo.Properties["Name"].
                get_Value();
        }
    }
}

```

## 6 Wnioski

Podczas ćwiczenia poznaliśmy podstawową obsługę skanera i funkcje takie jak zmiana parametrów skanowania, zapis do pliku, podgląd, czy wywołanie domyślnego interfejsu systemu Windows. Biblioteka WIA 2.0 okazała się intuicyjna w użyciu i pozwoliła na sprawne wykonanie zadań.

## 7 Bibliografia

- Dokumentacja Microsoftu:  
*[https : //docs.microsoft.com/en-us/windows/desktop/wia/-wia-startpage](https://docs.microsoft.com/en-us/windows/desktop/wia/-wia-startpage)*