

Wykorzystanie środowiska Visual Studio w procesie dydaktycznym nauki programowania robotów przemysłowych



*Politechnika Lubelska, Wydział Mechaniczny,
Katedra Informatyzacji i Robotyzacji Produkcji*

Łukasz SOBASZEK

Uzasadnienie podjęcia tematu

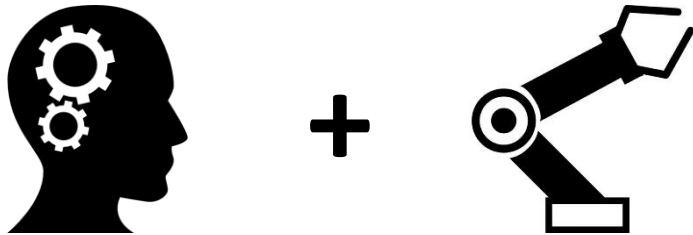
- W obszarze zautomatyzowanego wytwarzania oraz wspierania procesów produkcyjnych **wykorzystanie robotów** znajduje **coraz szersze zastosowanie**.
- Ich implementacja niesie za sobą **wiele korzyści** – m. in. zwiększenie wydajności i elastyczności produkcji, wzrost bezpieczeństwa, czy osiągnięcie wysokiej jakości produkcji.
- Roboty znajdują zastosowanie głównie na stanowiskach uciążliwych i niebezpiecznych, jednak coraz częściej **wykorzystuje się je** jako **urządzenia współpracujące z człowiekiem** lub jako **maszyny realizujące zasadniczą obróbkę** mechaniczną.
- Istotne jest zatem, aby **przyszli inżynierowie** posiadali **wiedzę z zakresu programowania oraz obsługi** tego typu urządzeń.



Programowanie robotów przemysłowych

Metody programowania robotów:

- **online** – wymagające obecności robota,
- offline – realizowane bez robota,
- hybrydowe – łączące powyższe metody.

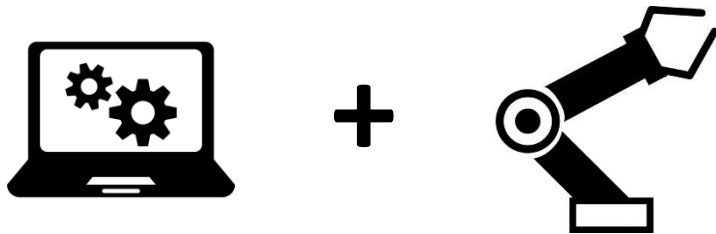


Rys. 1. Programowanie robota z wykorzystaniem Teach Pendant'a [źródło: *RobotWorx Blog*]

Programowanie robotów przemysłowych

Metody programowania robotów:

- online – wymagające obecności robota,
- **offline** – realizowane bez robota,
- hybrydowe – łączące powyższe metody.



Rys. 2. Wirtualne środowisko programowania robotów
[źródło: Octopuz Software]

Programowanie robotów przemysłowych

Metody programowania robotów:

- online – wymagające obecności robota,
- offline – realizowane bez robota,
- **hybrydowe** – łączące powyższe metody.



Rys. 3. Hybrydowe programowanie robota
[źródło: *Robotic Industries Association*]

Środowiska programowania robotów

The logo for ABB, consisting of the letters 'ABB' in a bold, red, sans-serif font.The logo for KUKA, consisting of the letters 'KUKA' in a bold, orange, sans-serif font.The logo for YASKAWA, consisting of the letters 'YASKAWA' in a bold, blue, sans-serif font.The logo for FANUC, consisting of the letters 'FANUC' in a bold, red, sans-serif font, set against a yellow rectangular background.The logo for EPSON, consisting of the letters 'EPSON' in a bold, blue, sans-serif font, with a registered trademark symbol (®) to the upper right.The logo for Kawasaki, featuring a stylized red 'K' symbol above the word 'Kawasaki' in a bold, black, sans-serif font.

Zalety proponowanych rozwiązań:

- szeroka gama modeli robotów, chwytaków i urządzeń pomocniczych,
- wierne odwzorowanie środowiska pracy robota,
- korzystanie z wirtualnego kontrolera oraz zdalne programowanie robotów,
- wbudowane kreatory zrobotyzowanych procesów produkcyjnych,
- możliwość przeprowadzenia analiz pracy robota – wykrywanie kolizji, szacowanie czasu pracy, dokonywanie pomiarów odległości i wiele innych.

Ograniczenia proponowanych rozwiązań



OBSŁUGA ŚRODOWISKA



DOSTĘPNOŚĆ LICENCJI



BŁĘDY OPROGRAMOWANIA



DOSTĘPNOŚĆ ROBOTA

Ograniczenia proponowanych rozwiązań



OBSŁUGA ŚRODOWISKA



DOSTĘPNOŚĆ LICENCJI



BŁĘDY OPROGRAMOWANIA



DOSTĘPNOŚĆ ROBOTA

Ograniczenia proponowanych rozwiązań



OBSŁUGA ŚRODOWISKA



DOSTĘPNOŚĆ LICENCJI



BŁĘDY OPROGRAMOWANIA



DOSTĘPNOŚĆ ROBOTA

Ograniczenia proponowanych rozwiązań



OBSŁUGA ŚRODOWISKA



DOSTĘPNOŚĆ LICENCJI



BŁĘDY OPROGRAMOWANIA



DOSTĘPNOŚĆ ROBOTA

Ograniczenia proponowanych rozwiązań



OBSŁUGA ŚRODOWISKA



DOSTĘPNOŚĆ LICENCJI

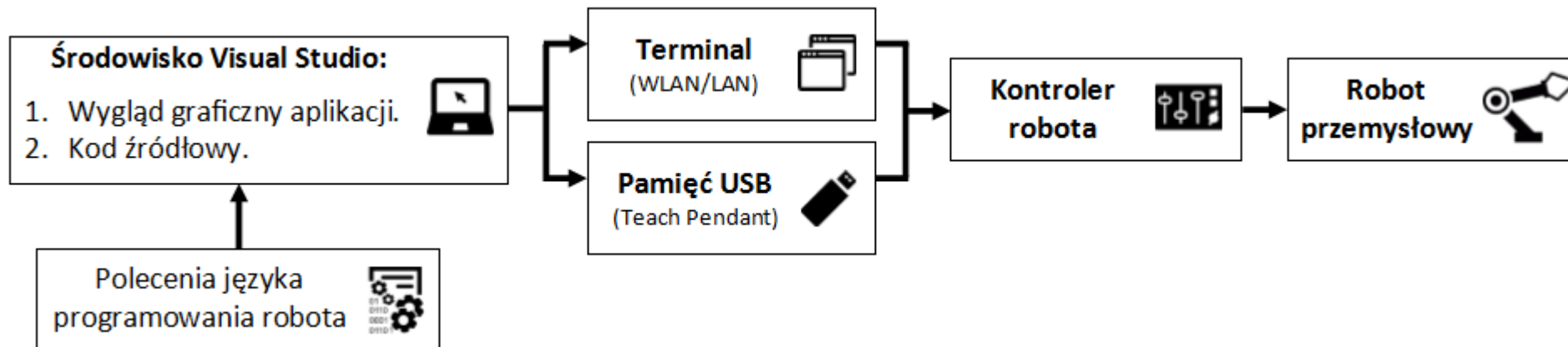


BŁĘDY OPROGRAMOWANIA



DOSTĘPNOŚĆ ROBOTA

Języki programowania robotów wywodzą się w dużej mierze z języków programowania komputerów...



Rys. 4. Wykorzystanie środowiska Visual Studio w procesie programowania robotów przemysłowych

Charakterystyka środowiska Visual Studio

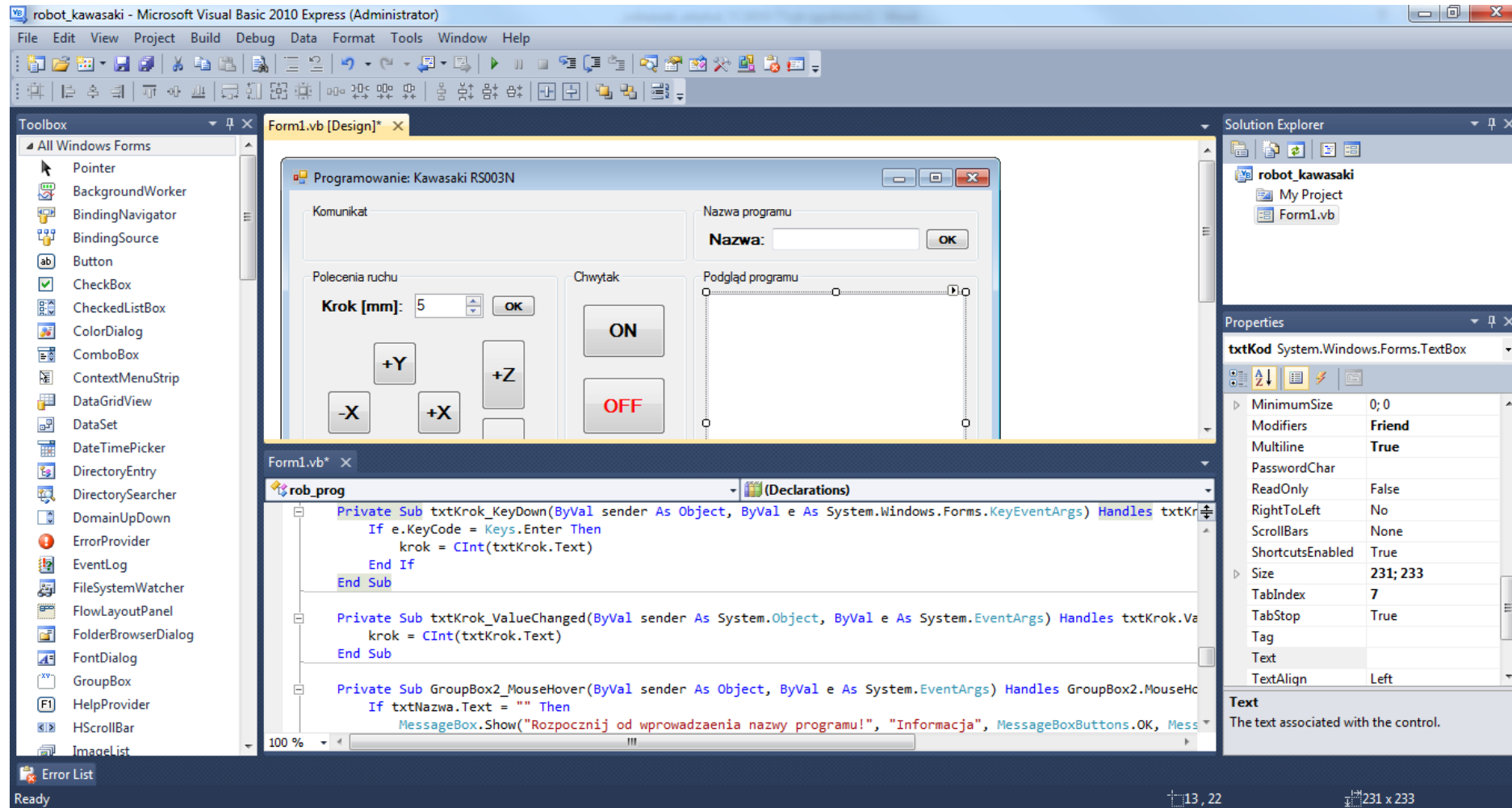
Visual Studio to wszechstronne, zintegrowane **środowisko programistyczne** firmy Microsoft, przeznaczone do tworzenia **zróżnicowanych aplikacji**. Niewątpliwą zaletą środowiska jest sam **sposób budowy aplikacji** (uprzednie przygotowanie projektu graficznego, a następnie kodowanie), intuicyjność, a także wbudowany interpreter.

Cechy charakterystyczne środowiska:

- jeden zintegrowany interfejs użytkownika (IDE) dla kilku języków i różnych typów projektów,
- połączenie z przeglądarką internetową,
- wbudowane narzędzia do debugowania,
- dostosowany do oczekiwań użytkownika interfejs.



Charakterystyka środowiska Visual Studio



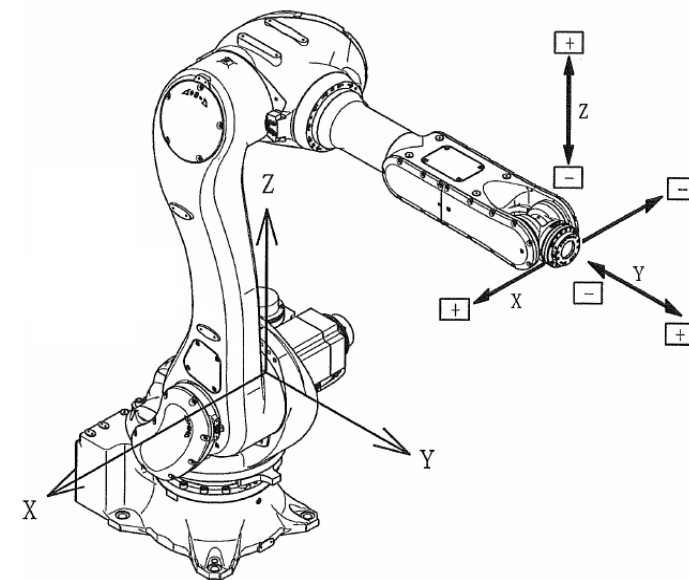
Rys. 5. Interfejs użytkownika w środowisku Visual Studio

Aplikacja wspomagająca programowanie robota przemysłowego

Celem prac było opracowanie aplikacji komputerowej umożliwiającej generowanie kodów sterujących robotem przemysłowym Kawasaki RS003N.

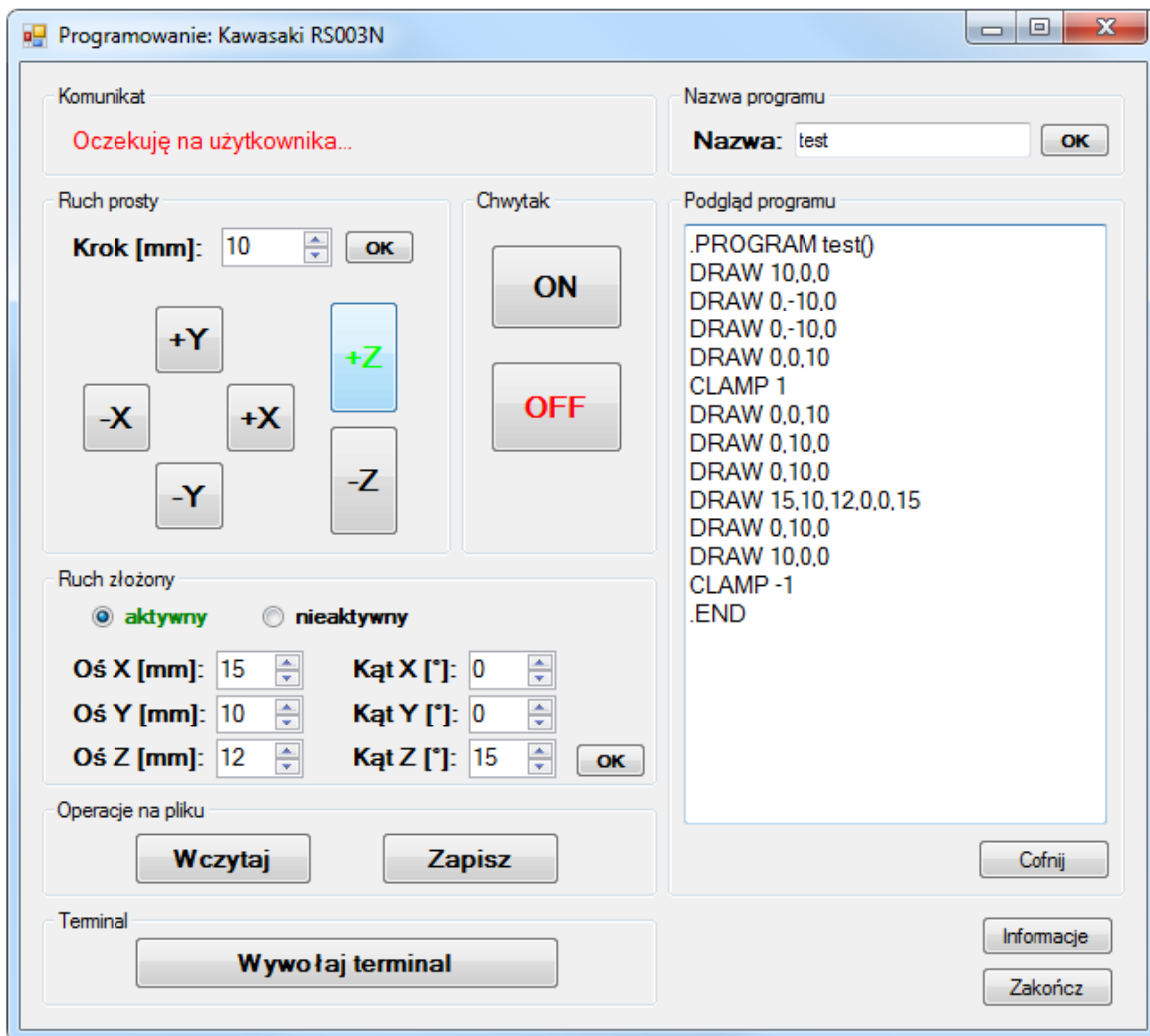
Założenia:

- opracowywana aplikacja zostanie wykorzystana podczas nauki podstaw programowania robota,
- ruch robota odbywa się zgodnie z układem współrzędnych osi narzędzia (tzw. układzie *TOOL*),
- punkty ruchu robota nie są uprzednio deklarowane (zastosowanie znajdują jedynie proste polecenia ruchu po współrzędnych),
- program powinien umożliwiać także złożone ruchy robota (obroty osi) oraz obsługę chwytaka podciśnieniowego.



Rys. 6. Przyjęty układ współrzędnych robota

Aplikacja wspomagająca programowanie robota przemysłowego



Rys. 7. Interfejs graniczny opracowanej aplikacji



Rys. 8. Wykorzystane stanowisko dydaktyczne

Aplikacja wspomagająca programowanie robota przemysłowego

```
Private Sub btnChwytaKON_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnChw
    txtKod.Text = txtKod.Text & "CLAMP 1" & vbCrLf
    btnChwytaKON.ForeColor = Color.Green
    btnChwytaOFF.ForeColor = Color.Black
    txtChwytaInfo.Visible = True
End Sub
```

```
Private Sub btnNazwaOK_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnNazwa
    txtKod.Text = txtKod.Text & ".PROGRAM " & txtNazwa.Text & "()" & vbCrLf
End Sub
```

```
Private Sub btnX_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnX.Click
    txtKod.Text = txtKod.Text & "DRAW " & krok & ",0,0" & vbCrLf
    lblInfo.Text = "Oś X: przemieszczenie do przodu o " & krok & " mm."
End Sub
```

```
Private Sub btn_X_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_X.Click
    txtKod.Text = txtKod.Text & "DRAW -" & krok & ",0,0" & vbCrLf
    lblInfo.Text = "Oś X: przemieszczenie do tyłu o " & krok & " mm."
End Sub
```

Rys. 9. Fragment kodu źródłowego opracowanej aplikacji

Aplikacja wspomagająca programowanie robota przemysłowego

```
Private Sub btnChwytaKON_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnChw
    txtKod.Text = txtKod.Text & "CLAMP 1" & vbCrLf
    btnChwytaKON.ForeColor = Color.Green
    btnChwytaKOFF.ForeColor = Color.Black
    txtChwytaInfo.Visible = True
End Sub
```

```
Private Sub btnNazwaOK_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnNazwa
    txtKod.Text = txtKod.Text & ".PROGRAM " & txtNazwa.Text & "()" & vbCrLf
End Sub
```

```
Private Sub btnX_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btnX.Click
    txtKod.Text = txtKod.Text & "DRAW " & krok & ",0,0" & vbCrLf
    lblInfo.Text = "Oś X: przemieszczenie do przodu o " & krok & " mm."
End Sub
```

```
Private Sub btn_X_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles btn_X.Click
    txtKod.Text = txtKod.Text & "DRAW -" & krok & ",0,0" & vbCrLf
    lblInfo.Text = "Oś X: przemieszczenie do tyłu o " & krok & " mm."
End Sub
```

Rys. 9. Fragment kodu źródłowego opracowanej aplikacji

Podsumowanie i wnioski

- Roboty przemysłowe znajdują **szerokie zastosowanie we współczesnych procesach produkcyjnych**, przez co konieczne jest, aby osoby znajdujące zatrudnienie w firmach produkcyjnych posiadały **podstawową wiedzę z zakresu programowania** tego typu urządzeń.
- Wykorzystanie podczas nauki programowania robotów środowiska **Visual Studio** może znacznie **usprawnić ten proces**.
- Zaprezentowane środowisko programistyczne pozwala na opracowywanie aplikacji o ściśle **zdefiniowanych możliwościach i funkcjonalności**, które dedykowane są konkretnym modelom robotów.
- Należy podejmować prace nad budową **zaawansowanych aplikacji użytkowych** – narzędzi wspomagających prace robotów w ściśle sprecyzowanych procesach – np. spawaniu, malowaniu, pakowaniu.

Dziękuję za uwagę!

dr inż. Łukasz Sobaszek

*Politechnika Lubelska, Wydział Mechaniczny,
Katedra Informatyzacji i Robotyzacji Produkcji*

l.sobaszek@pollub.pl

Wykorzystanie środowiska Visual Studio w procesie dydaktycznym nauki programowania robotów przemysłowych



*Politechnika Lubelska, Wydział Mechaniczny,
Katedra Informatyzacji i Robotyzacji Produkcji*

Łukasz SOBASZEK