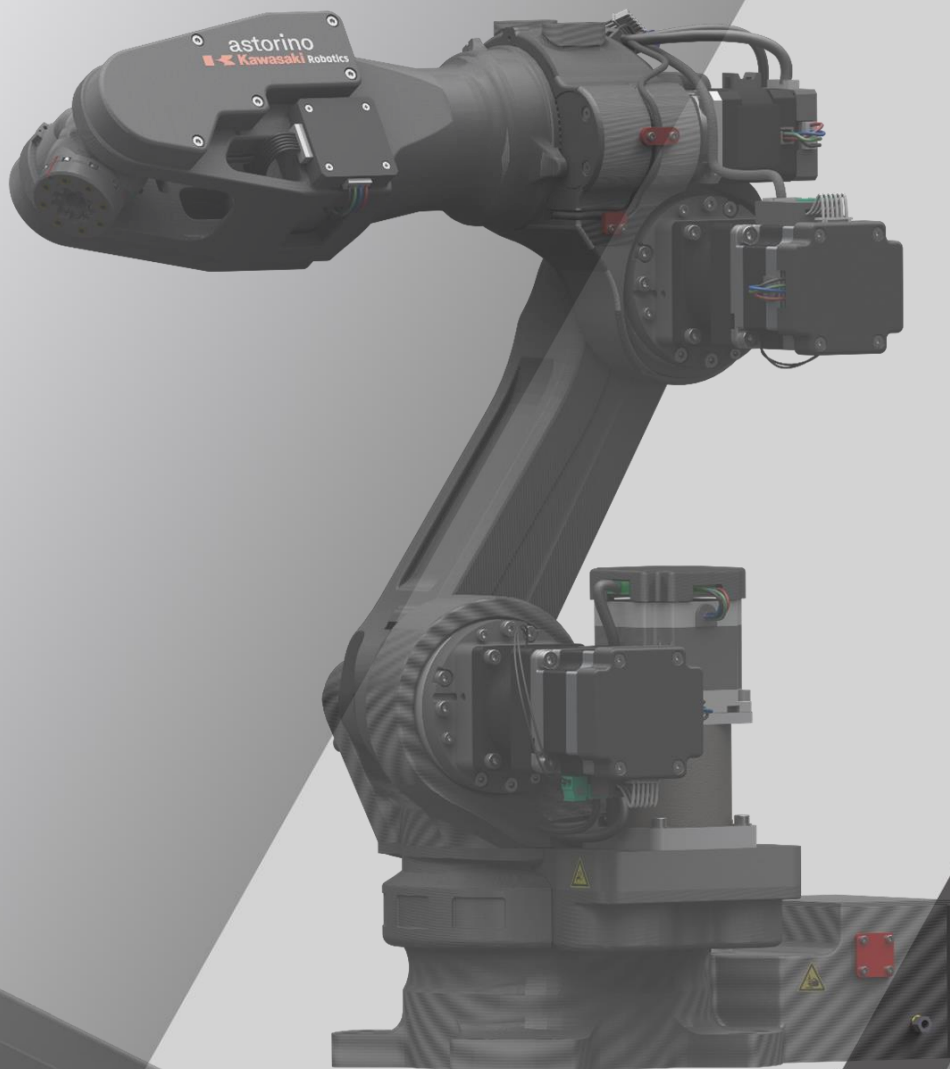


# Astorino

## Instrukcja obsługi





## **Wprowadzenie**

Niniejsza instrukcja opisuje obsługę 6-osioowego robota edukacyjnego "Kawasaki Robotics Astorino" i związanego z nim oprogramowania "Astorino", które jest częścią zakresu dostawy. **Instrukcja jest ważna od wersji firmware 3.8.4 oraz wersji oprogramowania astorino 1.9.2.**

ASTORINO to robot edukacyjny, który został opracowany specjalnie dla placówek i instytucji szkoleniowych. Uczniowie i studenci mogą korzystać z ASTORINO, aby uczyć się automatyzacji i robotyzacji procesów przemysłowych w praktyce.

W przypadku dalszych pytań skontaktuj się z pomocą techniczną ASTOR.

### **Kontakt:**

**Pomoc Techniczna ASTOR, Dział Osprzętu Robotyki**

**e-mail: [Astorino@astor.com.pl](mailto:Astorino@astor.com.pl)**

## ASTORINO Instrukcja obsługi

---

1. Oprogramowanie "Astorino" dołączone do Astorino jest licencjonowane wyłącznie do użytku z tym robotem i nie może być używane, kopiowane ani rozpowszechniane w żadnym innym środowisku.
  2. ASTOR i Kawasaki Robotics nie ponoszą odpowiedzialności za wypadki, szkody i/lub problemy spowodowane niewłaściwym użytkowaniem robota Astorino.
  3. ASTOR i Kawasaki Robotics zastrzegają sobie prawo do zmiany, poprawienia lub aktualizacji niniejszego podręcznika bez uprzedniego powiadomienia.
  4. Ta instrukcja nie może być przedrukowywana lub kopiowana w całości lub w części bez uprzedniej pisemnej zgody ASTOR i Kawasaki Robotics.
  5. Przechowuj tę instrukcję w bezpiecznym miejscu i w zasięgu ręki, aby można było z niej korzystać w dowolnym momencie. Jeśli instrukcja zostanie zgubiona lub poważnie uszkodzona, skontaktuj się z ASTOR.
- 

Copyright © 2024 ASTOR & Kawasaki Robotics GmbH.

Wszelkie prawa zastrzeżone.



## Symbole

Elementy, które wymagają szczególnej uwagi w tym podręczniku, są oznaczone następującymi symbolami.

Zapewnij prawidłowe działanie robota i zapobiegaj obrażeniom lub uszkodzeniom mienia, postępując zgodnie z instrukcjami bezpieczeństwa w polach z tymi symbolami.



### Ostrzeżenie

**Nieprzestrzeganie poniższej instrukcji może spowodować obrażenia.**

### [UWAGA]

Określa środki ostrożności dotyczące specyfikacji robota, obsługi, nauczania i konserwacji.



### Ostrzeżenie

- 1. Dokładność i skuteczność wykresów, procedur i wyjaśnień zawartych w niniejszym podręczniku nie może być potwierdzona z absolutną pewnością. W przypadku wystąpienia jakichkolwiek problemów z należy skontaktować się z Kawasaki Robotics GmbH lub z firmą Astor pod powyższym adresem.**
- 2. Aby upewnić się, że wszystkie prace są wykonywane bezpiecznie, przeczytaj ze zrozumieniem instrukcję. Ponadto należy zapoznać się ze wszystkimi obowiązującymi przepisami prawa, regulacjami i powiązanymi materiałami, a także z oświadczeniami dotyczącymi bezpieczeństwa opisanymi w każdym rozdziale. Przygotuj odpowiednie środki bezpieczeństwa i procedury dotyczące rzeczywistej pracy.**

## Parafrazy

W tym podręczniku są używane następujące reguły pisowni:

- W przypadku konkretnego naciśnięcia odpowiedni jest ujęty w nawiasy kątowe, np. <F1> lub <Enter>.
- W przypadku przycisku okna dialogowego lub paska narzędzi nazwa przycisku jest ujęta w nawiasy kwadratowe, np. [OK] lub [Resetuj].
- Pola do wyboru są oznaczone kwadratowym polem. Jeśli są one aktywowane, wewnątrz  symbolu znajduje się również mały znacznik  wyboru.

## ASTORINO Instrukcja obsługi

---

### Spis treści

Wprowadzenie .....	I
Symbole.....	1
Parafrazy .....	2
Spis treści .....	3
1 Nazewnictwo w tym podręczniku.....	7
1.1 Opis robota ASTORINO .....	7
2 Specyfikacja techniczna .....	8
3 Zakres dostawy .....	9
4 Zakres ruchu .....	10
5 Wymiary montażowe .....	11
6 Punkty montażowe akcesoriów .....	12
7 Wykres obciążenia .....	15
8 Połączenia elektryczne.....	16
9 Uwagi dotyczące bezpieczeństwa .....	17
10 Rozpakowywanie i uruchamianie .....	18
10.1 Podłączanie akcesoriów .....	18
10.2 Wymagania systemowe.....	20
10.3 Instalacja sterownika .....	20
10.4 Instalacja oprogramowania Astorino .....	21
10.5 Przygotowanie Astorino do pracy .....	22
11 Układy współrzędnych .....	25
11.1 Układ współrzędnych BASE .....	25
11.2 Układ współrzędnych JOINT .....	26
12 Tryby pracy robota .....	27
12.1 Tryb nauczania (Teach).....	27
12.2 Tryb powtarzania (Repeat) .....	27
13 Ręczna obsługa robota .....	28
13.1 JOINT (Ruch złączowy) .....	28
13.2 BASE (Ruch względem podstawowego układu wsp.) .....	29
13.3 TOOL (Ruch względem układu narzędzia) .....	30
13.4 WORK.....	32
14 Ruch robota.....	33
14.1 Interpolacja liniowa .....	34
14.2 Interpolacja złączowa .....	34
14.3 Interpolacja kołowa .....	35
15 Oprogramowanie Astorino .....	36
15.1 Podstawowe informacje.....	36
15.2 Obszar Status .....	37
15.3 Karta Control / Sterowanie .....	38
15.3.1 Obszar Motors / Napędy .....	39

## ASTORINO Instrukcja obsługi

15.3.2	Obszar Control / Sterowanie .....	41
15.3.3	Obszar Connection / Połączenie .....	41
15.4	Karta JOG.....	42
15.4.1	Obszar Jogging.....	45
15.4.2	Obszar Current Position / Obecna Pozycja.....	47
15.4.3	Obszar Step - Teach / Ustawienia kroku .....	47
15.4.4	Obszar Teach Point / Naucz punkt .....	48
15.4.5	Obszar Execute Motion Command / Wykonaj Polecenie Ruchu .....	48
15.5	Karta Points / Punkty .....	49
15.6	Karta Home/Tool / Pozycja domowa/Tool .....	50
15.6.1	Obszar Tool .....	51
15.6.2	Obszar Wizard / Kreator .....	51
15.6.3	WORK .....	52
15.6.4	WORK WIZARD .....	52
15.6.5	Obszar Power OFF position / Pozycja wyłączenia.....	53
15.6.6	Obszar Zeroing order / Kolejność zerowania .....	53
15.7	Karta Moving Area / Przestrzeń Robocza .....	54
15.7.1	XYZ Limits .....	54
15.7.2	Range .....	55
15.7.3	Working Space .....	56
15.8	Karta Program .....	57
15.9	Karta System Settings / Ustawienia Systemu.....	60
15.10	Obszar Calibration / Kalibracja.....	61
15.11	Pole Terminal .....	61
15.11.1	Obszar Status i konfiguracja .....	62
15.11.1.1	IO .....	62
15.11.1.2	MODBUS .....	63
15.11.1.3	Clamp.....	63
15.11.1.4	Sygnaly dedykowane .....	64
15.11.1.5	Detekcja kolizji .....	64
15.11.1.6	Taśmociąg.....	65
15.11.1.7	Ethernet.....	66
15.11.1.8	Firmware.....	66
15.12	Okno Wizualizacji.....	67
15.12.1	Obsługa okna wizualizacji .....	67
15.12.2	Klasy obiektów .....	68
15.12.3	Generator prostych kształtów.....	69
15.12.4	Menu modyfikacji obiektów .....	71
15.12.5	Menu ustawień wizualizacji .....	72
15.12.6	Wizualizacja Working Space .....	73
15.12.7	Wizualizacja zasięgu pracy robota.....	73
15.13	Karta About / Informacje .....	74

## ASTORINO Instrukcja obsługi

15.14	Aktualizacja oprogramowania sprzętowego.....	75
15.14.1	Podstawowe informacje .....	75
15.14.2	Procedura aktualizacji .....	76
15.14.3	Odzyskiwanie po awarii aktualizacji.....	78
15.15	Język AS.....	80
15.16	Programowanie.....	84
15.16.1	Tworzenie nowego programu .....	84
15.16.2	Napisz swój własny program .....	85
15.16.3	Ładowanie programu do robota .....	85
15.16.4	Uruchamianie aktualnie wybranej linijki.....	86
15.16.5	Uruchamianie programu .....	87
15.16.6	Zatrzymywanie programu.....	88
16	Przykładowe programy.....	89
16.1	Pick&Place – przykład paletyzacji.....	89
16.2	Przykładowy program we/wy .....	91
16.3	Przykładowy program do komunikacji szeregowej.....	91
17	Dane narzędzia .....	93
17.1	Dane narzędzia ze znanych wymiarów .....	93
17.2	Automatyczna konfiguracja narzędzia (współrzędne) .....	95
17.2.1	Przegląd funkcji automatycznej konfiguracji narzędzi .....	95
17.2.2	Dane wymagane do automatycznej konfiguracji współrzędnych narzędzia .....	96
17.2.3	Nauczanie czterech podstawowych pozycji.....	96
17.2.4	Nauczanie sześciu podstawowych pozycji.....	99
18	Automatyczna konfiguracja układu WORK (WSPÓŁRZĘDNE) .....	104
18.1.1	Uczenie trzech podstawowych punktów .....	104
19	Auto-kalibracja detekcji kolizji .....	106
20	Wejścia/wyjścia – 3,3V .....	107
20.1	Adapter I/O 3.3V.....	109
20.1.1	Adapter I/O 3.3V – montaż.....	109
21	Wyjścia oraz wejścia na ramieniu robota .....	110
22	Moduł I/O 24V .....	111
22.1	Podłączenie do robota .....	112
22.2	Podłączanie wejść.....	112
22.3	Podłączanie wyjść.....	113
23	MODBUS TCP.....	114
23.1	Tryby pracy sieci Modbus .....	114
23.2	Typy obiektów Modbus w Astorino .....	115
23.3	Konfiguracja portu Ethernet.....	116
23.4	Panel ASTRAADA HMI – przykład.....	117
23.5	Używanie rejestrów modbus jako wartości rzeczywistych.....	122
24	Kalibracja.....	123
25	Informacje o producencie.....	124

## ASTORINO Instrukcja obsługi

---

Załącznik A – Standardowa procedura zerowania .....	125
Załącznik B – materiał PET-G .....	127
Załącznik 3 – połączenie typu PNP .....	128
Załącznik 4 – Teensy 4.1 .....	129

## 1 Nazewnictwo w tym podręczniku

W tej sekcji znajdziesz definicje terminów używanych w tym podręczniku.

Autor podręcznika stara się posługiwać ogólnie obowiązującą terminologią, zachowując jak największą logikę. Niestety, należy zauważyć, że postrzeganie użytej terminologii może być różne w zależności od punktu widzenia, nawet przy rozważaniu tego samego tematu. Należy również stwierdzić, że w toku historii rozwoju robotów, komputerów i oprogramowania terminologia rozwijała się w różny sposób. W nowoczesnej instrukcji nie znajdziemy zatem terminologii, która zawsze będzie zgodna w 100% z opiniami wszystkich użytkowników i ekspertów.

### 1.1 Opis robota ASTORINO

ASTORINO to 6-osiovy robot edukacyjny z silnikami krokowymi pracującymi w pętli zamkniętej sterowania. Robot został opracowany specjalnie dla placówek i instytucji edukacyjnych, takich jak np. szkoły i uniwersytety.

Konstrukcja robota opiera się na druku 3D ze specjalnym włóknem węglowym. Korzystając z dostarczonych plików STL, można dodrukować uszkodzone części.

Programowanie i sterowanie odbywa się za pomocą oprogramowania "Astorino", które można znaleźć na dostarczonej pamięci USB, a najnowszą wersję można pobrać z serwera Kawasaki Robotics FTP:

<https://ftp.kawasakirobot.de/Software/Astorino/>

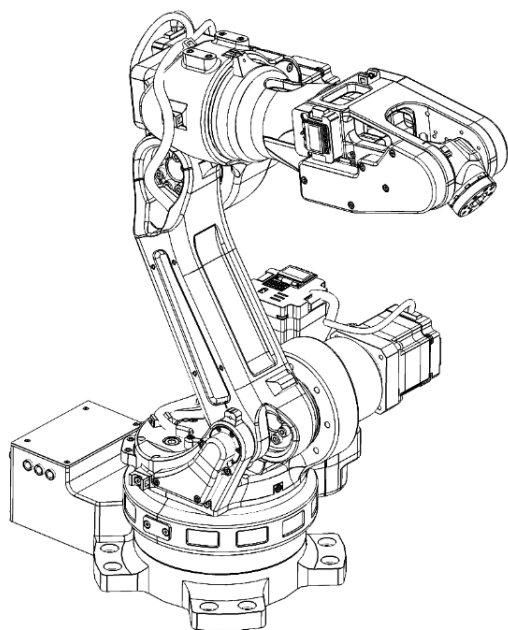
Podobnie jak roboty przemysłowe, robot Kawasaki Robotics Astorino jest zaprogramowany w języku AS i umożliwia użytkownikowi programowanie rzeczywistych aplikacji przemysłowych dla robotów Kawasaki Robotics.

## 2 Specyfikacja techniczna

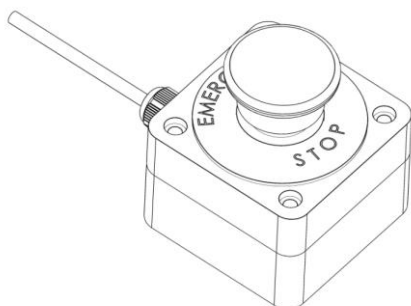
Charakterystyka		ASTORINO
Typ		Robot 6-osiowy
Maksymalny udźwig		1 kg
Liczba osi		6
Maksymalny zasięg		578 mm
Powtarzalność		±0,1 mm
Zakres ruchu	Oś 1 (JT1)	±158°
	Oś 2 (JT2)	-90°÷127°
	Oś 3 (JT3)	0°÷168°
	Oś 4 (JT4)	±240°
	Oś 5 (JT5)	±120°
	Oś 6 (JT6)	±360°
Prędkość pojedynczej osi	Oś 1 (JT1)	38°/s
	Oś 2 (JT2)	26°/s
	Oś 3 (JT3)	26°/s
	Oś 4 (JT4)	67,5°/s
	Oś 5 (JT5)	67,5°/s
	Oś 6 (JT6)	128.5°/s
Dopuszczalny moment	Oś 4 (JT4)	6,2 Nm
	Oś 5 (JT5)	1,45 Nm
	Oś 6 (JT6)	1,1 Nm
Środowisko pracy	Temperatura	0–40°C
	Wilgotność	35–80%
Kontroler		Teensy 4.1
Wejścia/wyjścia		8/8 (PNP 8 mA, NPN 15 mA)
		2/2 (24V PNP na JT3)
Maksymalny pobór prądu		144 W
Zasilanie		100–240 V, 50–60 Hz
Ciężar		12 kg
Pozycja montażowa		Podłogowy
Materiał		PET-G
Kolor		Czarny
Komunikacja		MODBUS TCP, TCP/IP, UDP, Serial
Wykrywanie kolizji		Akcelerometr
Bezpieczeństwo utraty zasilania		Hamulce w JT2 i JT3
Opcje	Moduł I/O 24V	8 × wejść/wyjść
	7. oś	Tor liniowy
	System wizyjny	OpenMV
	Śledzenie taśmy	2 enkodery



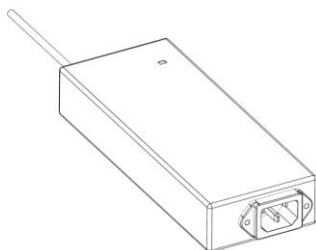
### 3 Zakres dostawy



Robot ASTORINO

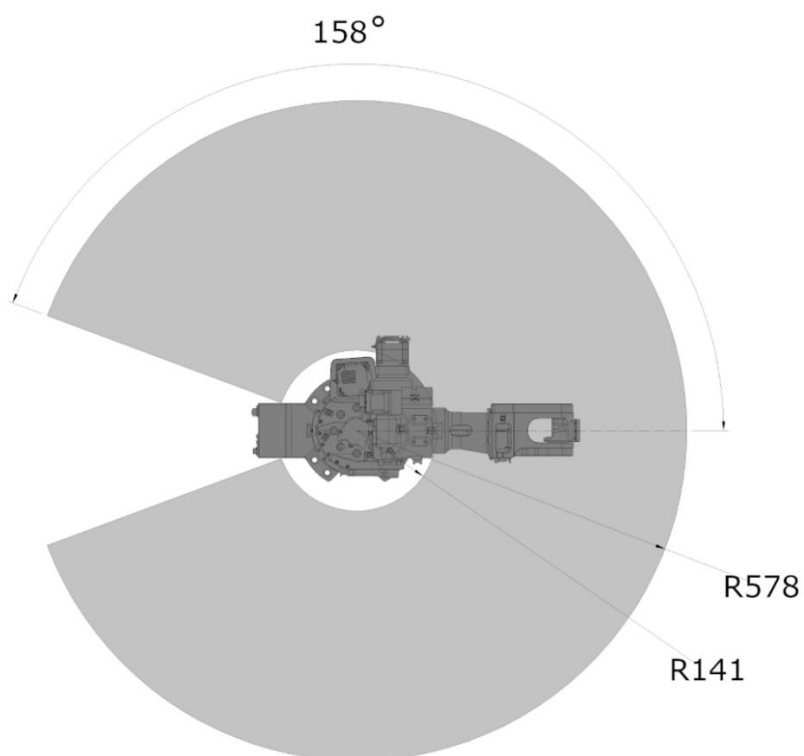
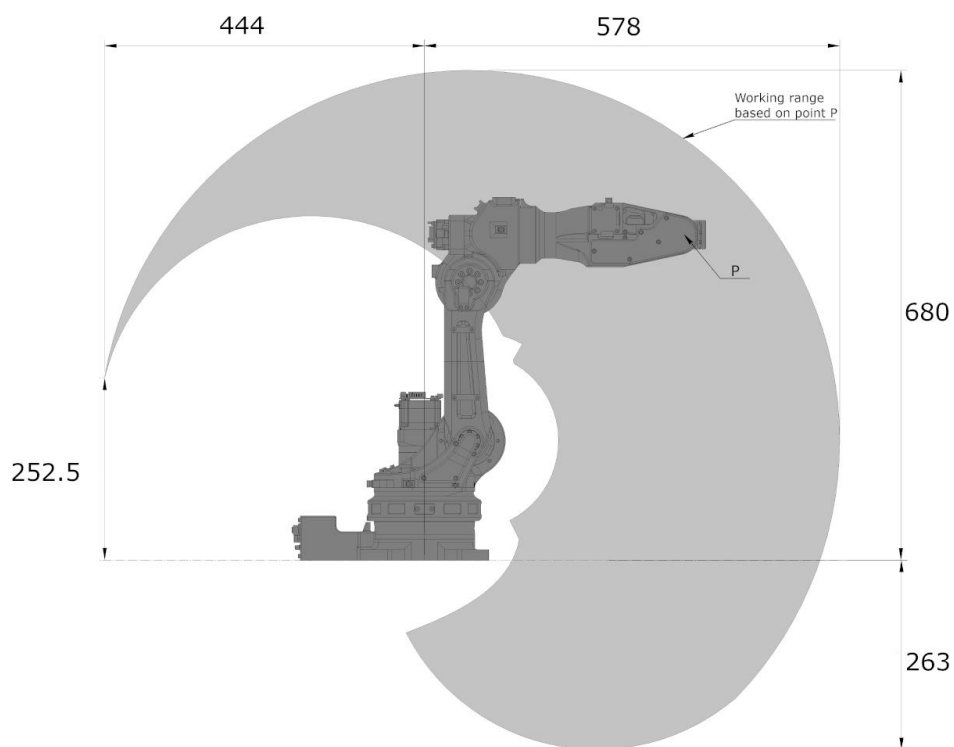


Przycisk bezpieczeństwa  
w obudowie

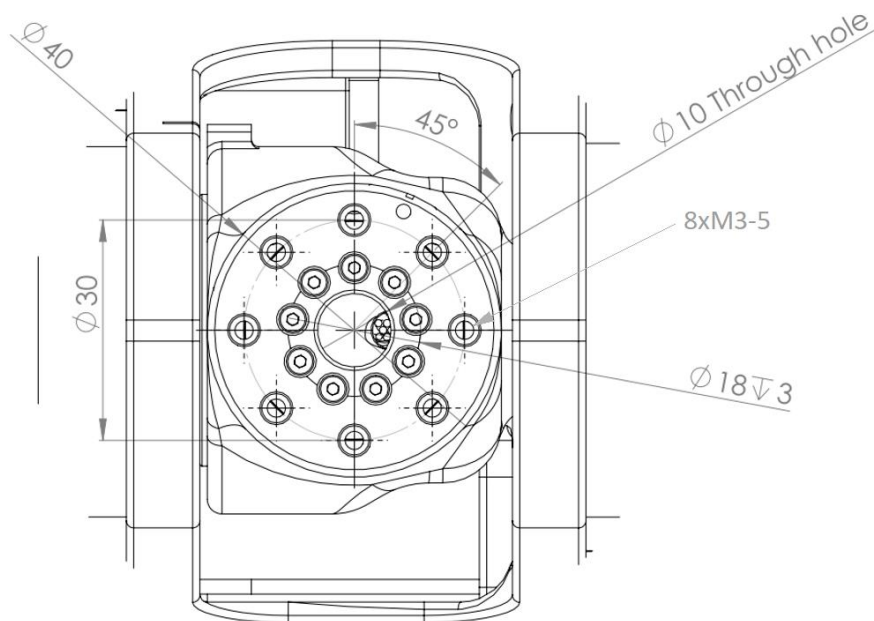
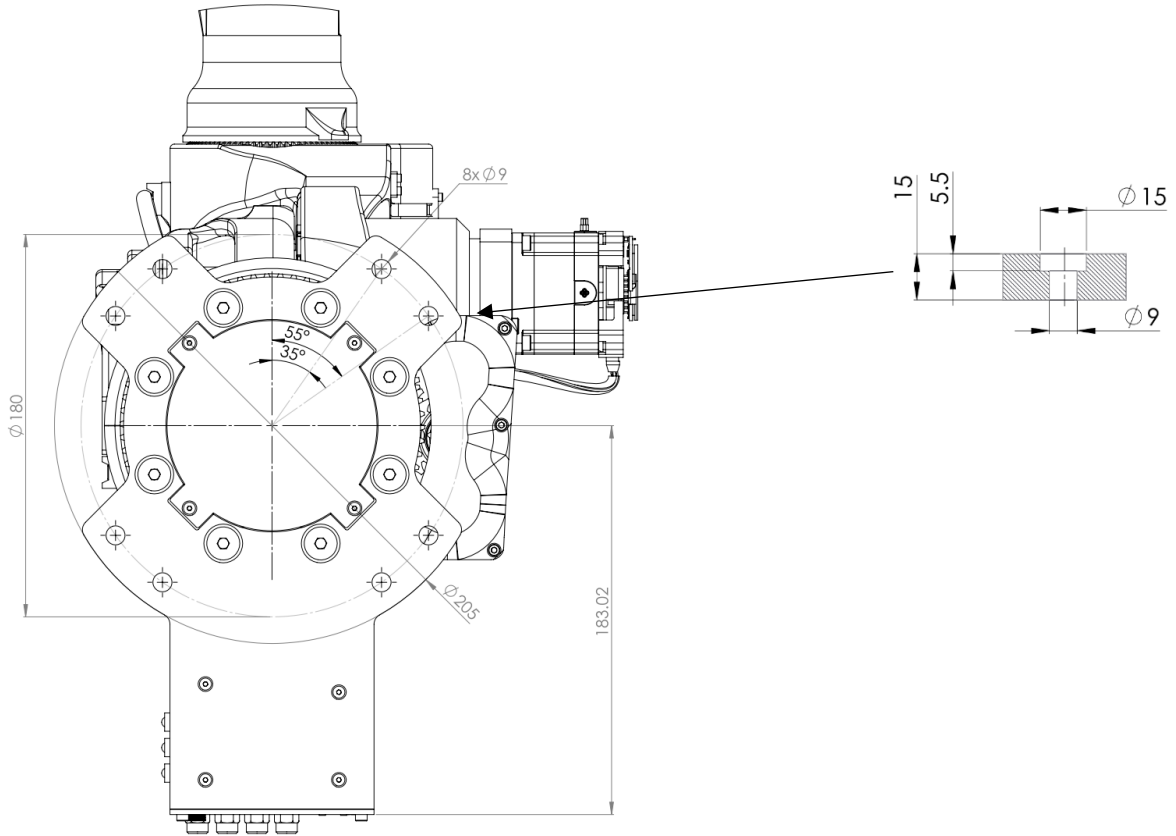


Zasilacz 24V/DC, przewód USB i  
pamięć USB

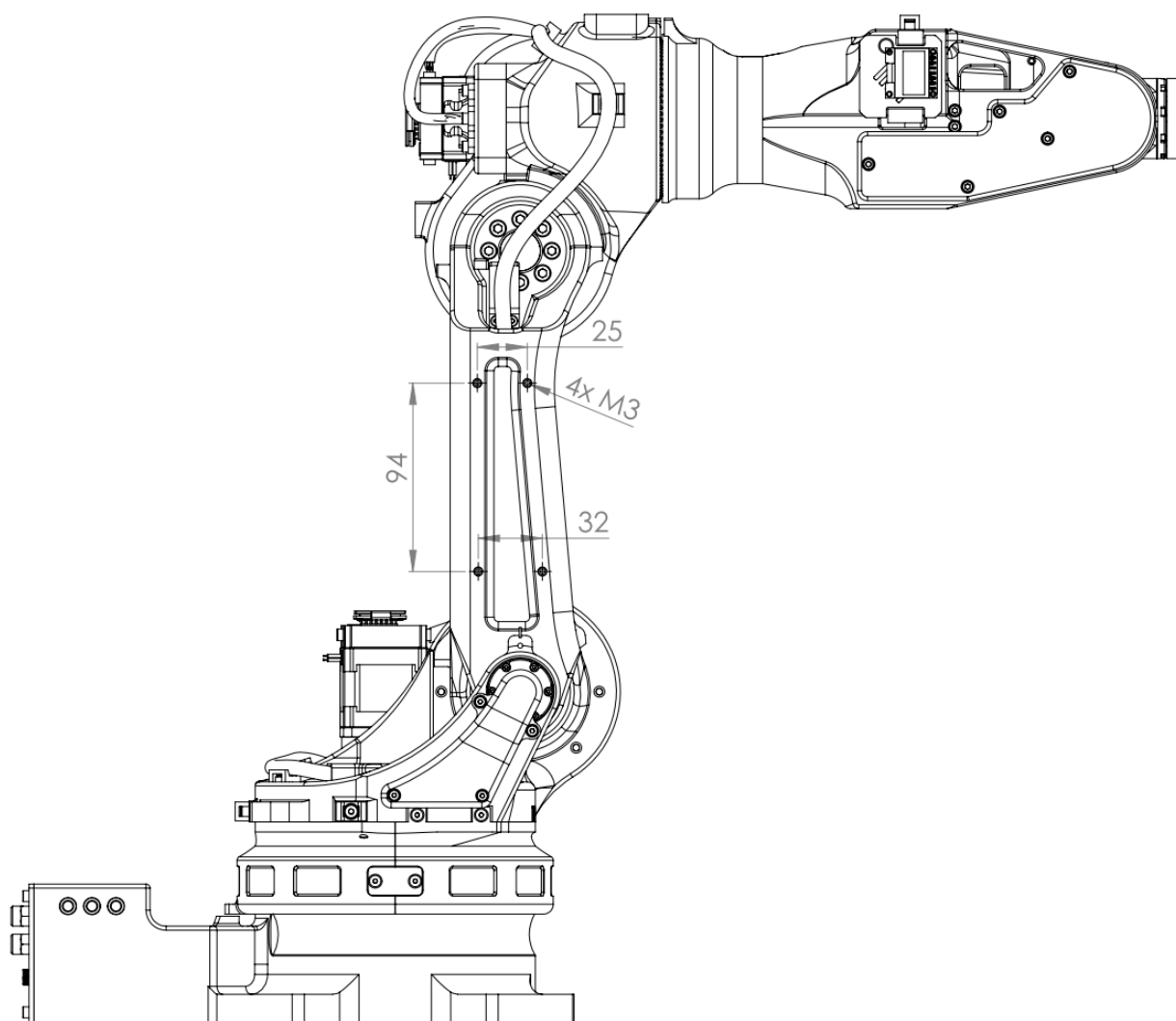
## 4 Zakres ruchu



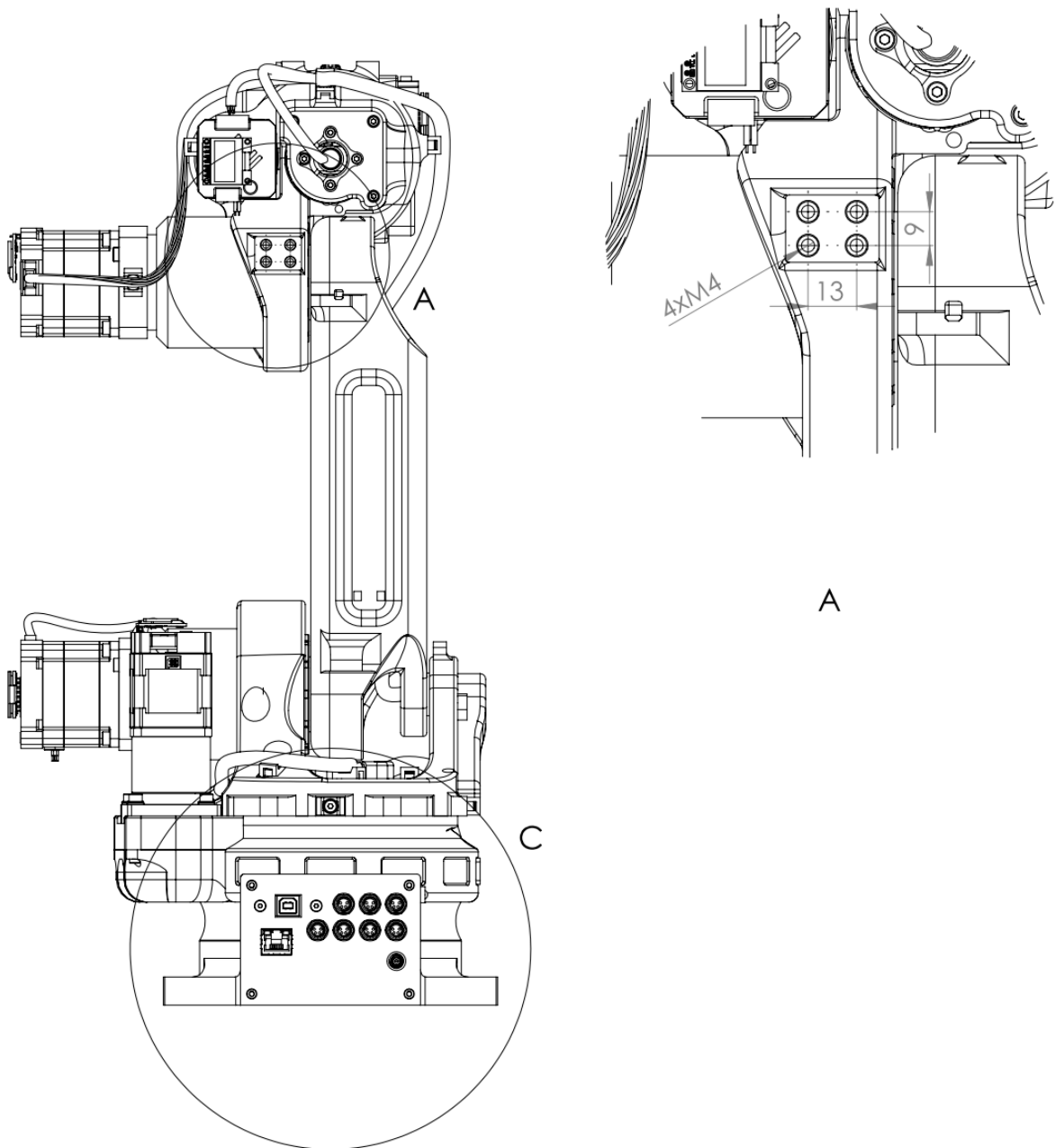
## 5 Wymiary montażowe



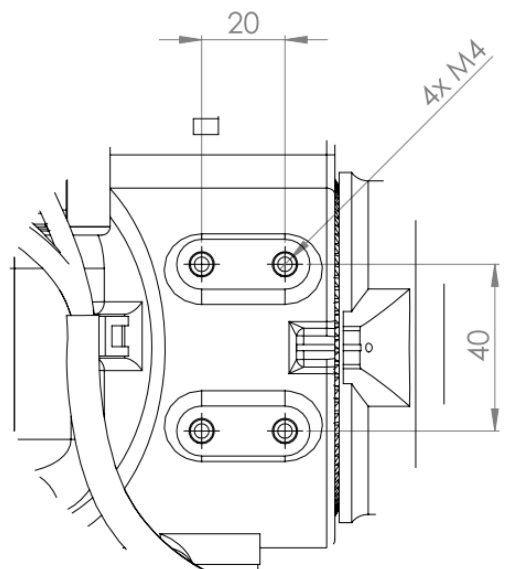
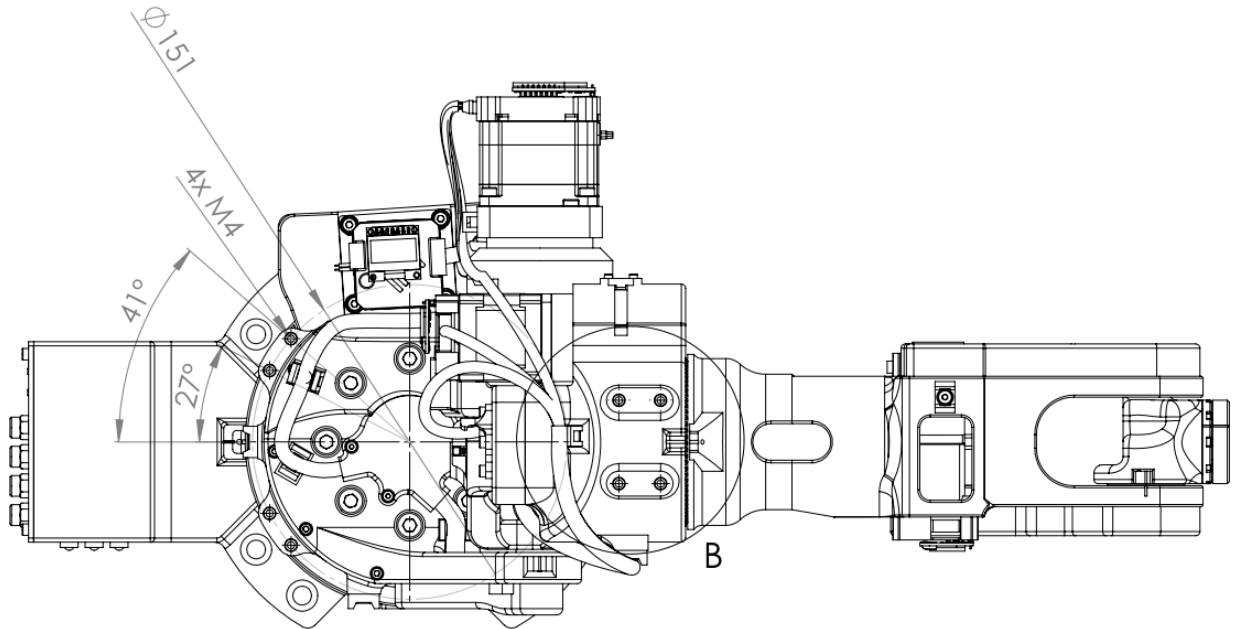
## 6 Punkty montażowe akcesoriów



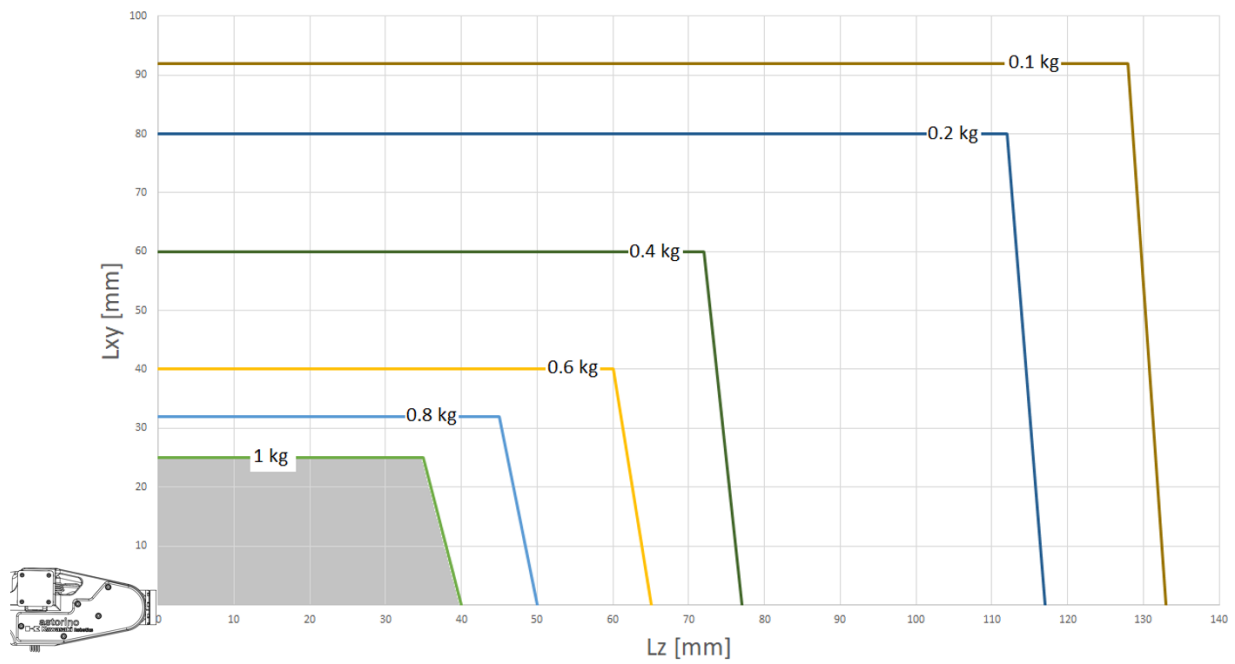
# ASTORINO Instrukcja obsługi



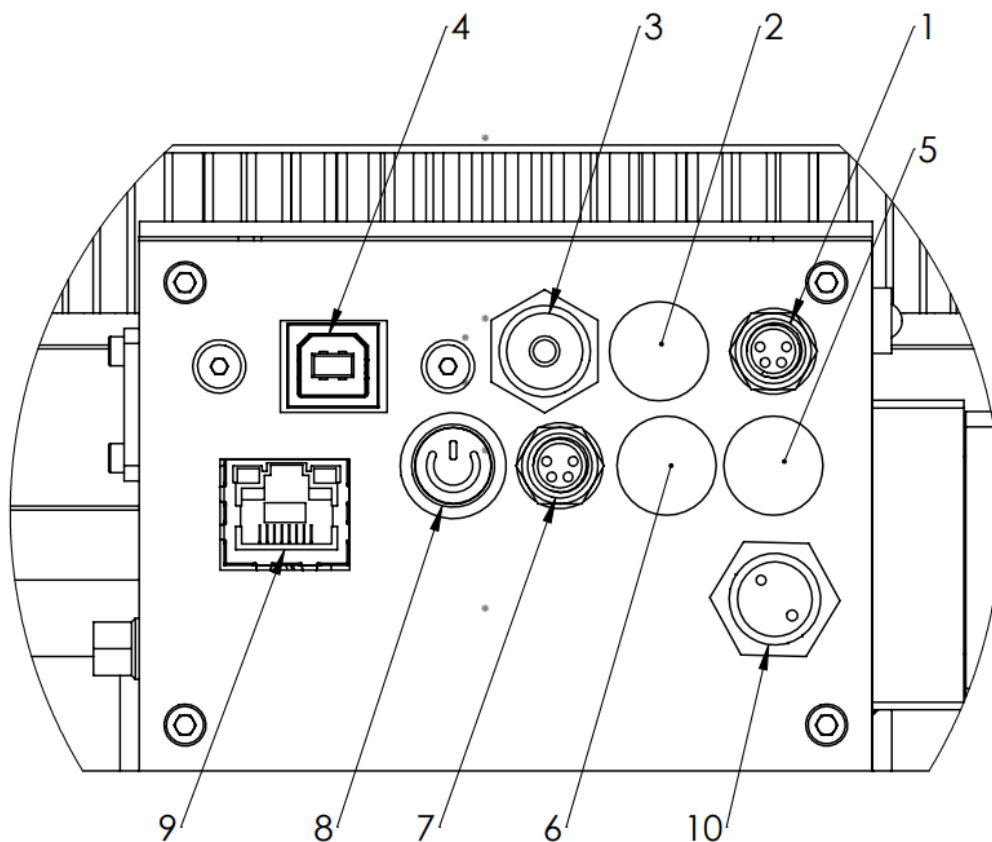
# ASTORINO Instrukcja obsługi



## 7 Wykres obciążenia



## 8 Połączenia elektryczne



1.	Gniazdo M8 4-pin – zewnętrzny wyłącznik awaryjny (E-Stop)
2.	Zewnętrzny wyłącznik awaryjny SAFETY-FENCE (OPCJA)
3.	Wlot ciśnienia Ø4,0 mm
4.	Port USB-B
5.	OPCJA 2 (enkoder 2 – przenośnik taśmowy 2/JT7)
6.	OPCJA 1 (enkoder 1 – przenośnik taśmowy 1)
7.	System wizyjny/Komunikacja szeregową (Serial) (Piny: 1-GND, 2 – 5V [Wizja], 3-TX, 4- RX)
8.	Przełącznik ON/OFF
9.	Port Ethernet (RJ45)
10.	Złącze zasilania



## 9 Uwagi dotyczące bezpieczeństwa

### [UWAGA]

Zawsze dbaj o bezpieczeństwo osobiste użytkowników i innych osób podczas obsługi ramienia robota lub uruchamiania celi robota!

- W wersji podstawowej robot nie posiada elementów związanych z bezpieczeństwem stanowiska zrobotyzowanego. W zależności od aplikacji może być konieczne ich dodanie. Podstawowa wersja robota wyposażona jest w przycisk awaryjny.
- Oznaczenie CE: Ramię robota podczas pracy w zastosowaniach produkcyjnych musi podlegać ocenie ryzyka i musi być zgodne z obowiązującymi przepisami bezpieczeństwa, aby zapewnić bezpieczeństwo osobiste. W zależności od wyniku oceny należy zintegrować dalsze elementy bezpieczeństwa. Są to zwykle przekaźniki bezpieczeństwa i wyłączniki drzwiowe. Odpowiedzialny jest inżynier rozruchu systemu. Aplikacje edukacyjne nie wymagają dodatkowych elementów bezpieczeństwa.
- Sterownik robota zawiera zasilacz 24 V, który sam wymaga napięcia sieciowego (100/240 V). Sprawdź etykietę na zasilaczu. Tylko wykwalifikowany personel może podłączyć zasilacz do sieci i uruchomić go.
- Prace przy elektronice robota powinny być wykonywane wyłącznie przez wykwalifikowany personel. Sprawdź aktualne wytyczne dotyczące wyładowań elektrostatycznych (ESD).
- Zawsze odłączaj robota od zasilania (100/240 V) podczas pracy w podstawie robota (sterowniku) lub jakiegokolwiek elektronice podłączonej do kontrolera robota.
- **NIE** podłączać na gorąco! Może to spowodować trwałe uszkodzenie modułów silnika. Nie należy instalować ani usuwać żadnych modułów ani złączy wtykowych/odłączających (np. przycisku zatrzymania awaryjnego, modułów DIO, złączy silnika) przy włączonym zasilaniu.
- Ramię robota musi być ustawione na stabilnej powierzchni i przykręcone lub zabezpieczone w inny sposób.
- Używaj i przechowuj robota wyłącznie w suchym i czystym środowisku.
- Używaj systemu wyłącznie w temperaturze pokojowej (15° do 32°C) - zalecane.

## 10 Rozpakowywanie i uruchamianie

Po wyjęciu robota z opakowania umieść go na solidnej powierzchni i zabezpiecz.

### 10.1 Podłączanie akcesoriów

- Należy przestrzegać instrukcji bezpieczeństwa.
- Zamontuj robota na odpowiedniej podstawie, stole lub metalowej płycie. Robot bez chwytaka lub innych akcesoriów samodzielnie wykonuje podstawowe operacje, takie jak zerowanie i uczenie ruchów w bezpośrednim sąsiedztwie podstawy robota. Ze względów bezpieczeństwa zalecamy przykręcenie podstawy robota przed włączeniem zasilania.
- Podłącz wtyczkę 2-stykową zasilacza i 4-stykową wtyczkę M8 ze wewnętrznego przycisku zatrzymania awaryjnego do połączeń na podstawie robota.



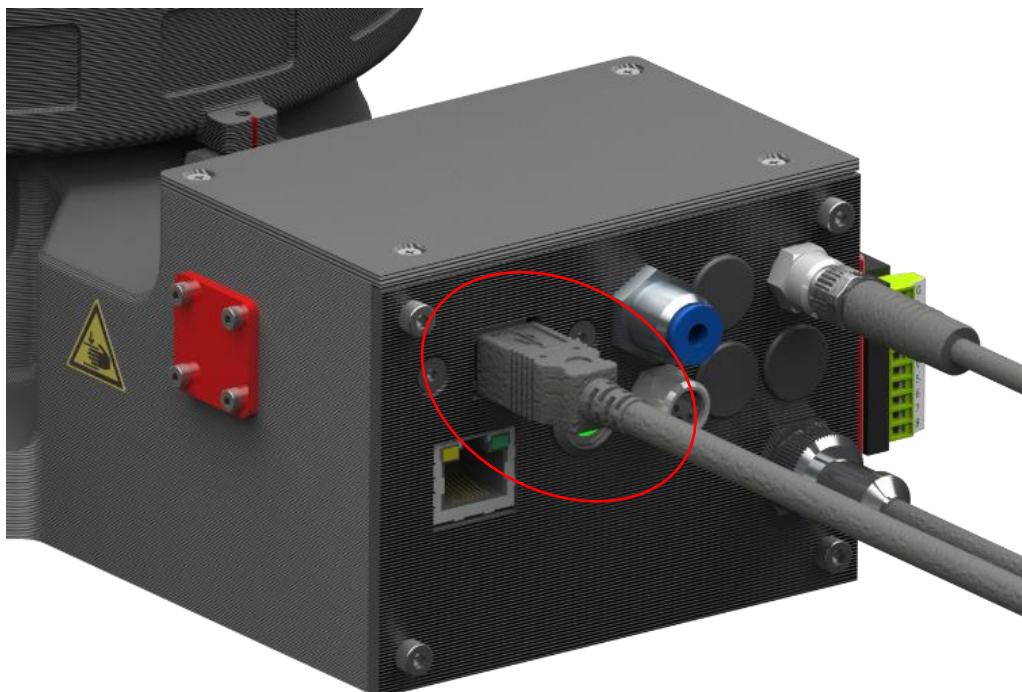
## ASTORINO Instrukcja obsługi

---

- Włącz zasilanie, naciskając podświetlony przycisk.



- Podłącz USB do portu USB-B podstawy robota, a następnie podłącz go do komputera.



## 10.2 Wymagania systemowe

Przed zainstalowaniem oprogramowania astorinoIDE upewnij się, że komputer spełnia następujące wymagania sprzętowe i programowe.

Część	Wymagania
CPU	2.0 Ghz lub szybszy
Pamięć	4 GB minimum
Dysk	100 MB wolnej przestrzeni
Karta graficzna	Dowolna
Ustawienia ekranu	Rozdzielczość minimalna 1280 x 720, zalecane skalowanie ekranu 100 %
Mysz	Trzy przyciskowa

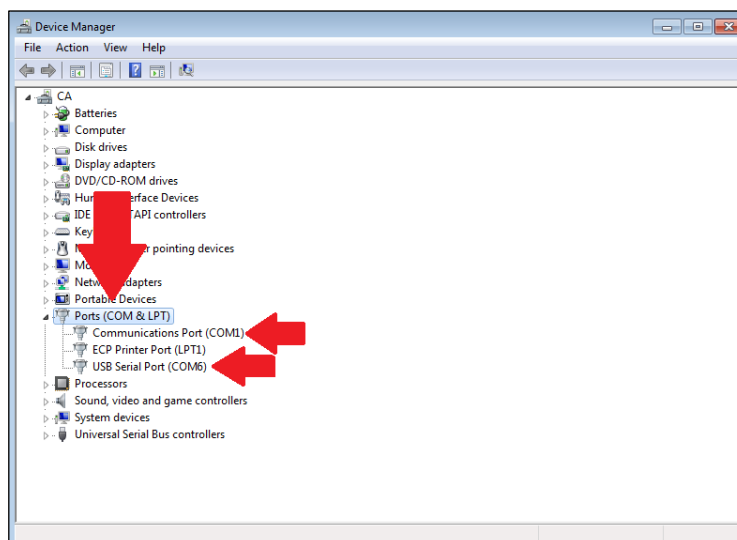
System	Wersja
Windows	7, 8, 8.1, 10, 11

## 10.3 Instalacja sterownika

System operacyjny instaluje wymagane sterowniki automatycznie od wersji Windows 8. Po udanej instalacji robot powinien pojawić się w Menedżerze urządzeń w pozycji <Ports>.

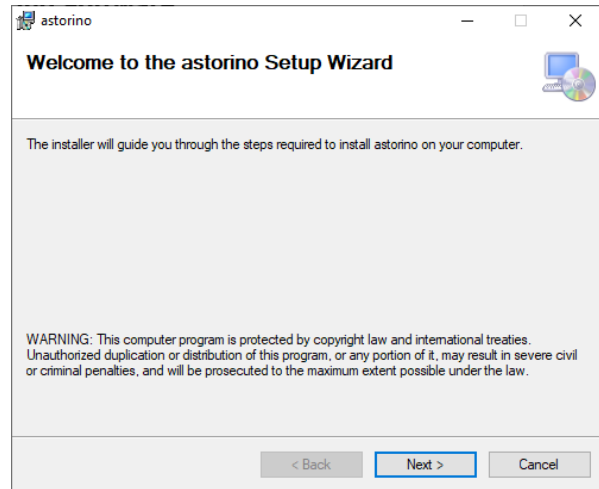
Jeśli korzystasz z systemu Windows 7, zainstaluj sterowniki przed podłączeniem robota do komputera (pobranego z serwera FTP Kawasaki lub z pamięci USB).

*Wywołanie menedżera urządzeń za pośrednictwem <Windows + R> devmgmt.msclub ⇒ kliknięcie ikony w menu wyboru za pośrednictwem <Windows + X>*

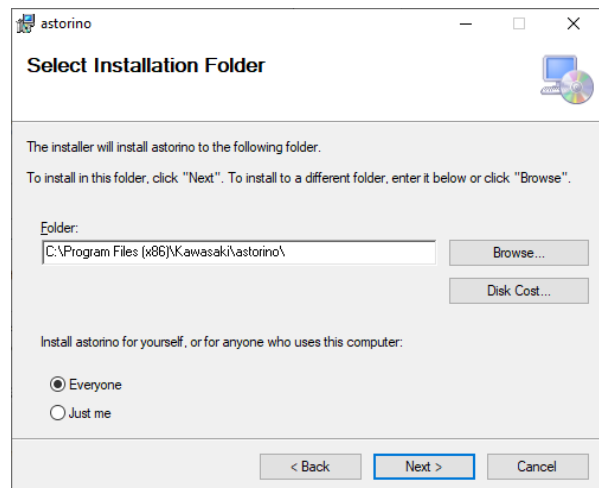


## 10.4 Instalacja oprogramowania Astorino

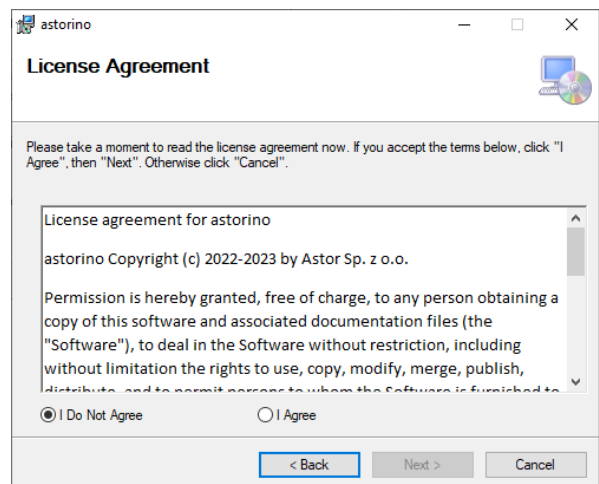
Uruchom Astorino\_x.x.x.exe



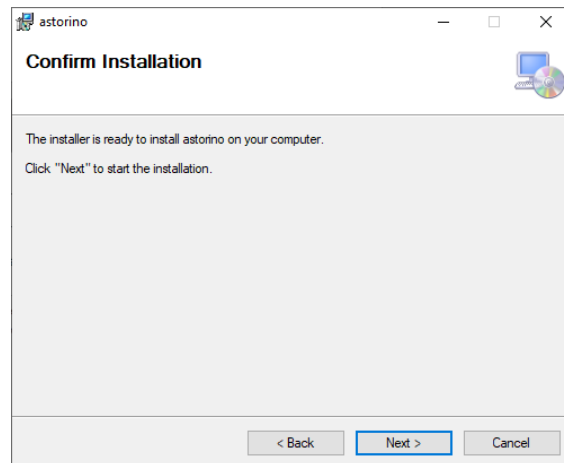
Potwierdź lub dostosuj katalog instalacyjny



Zaakceptuj licencję

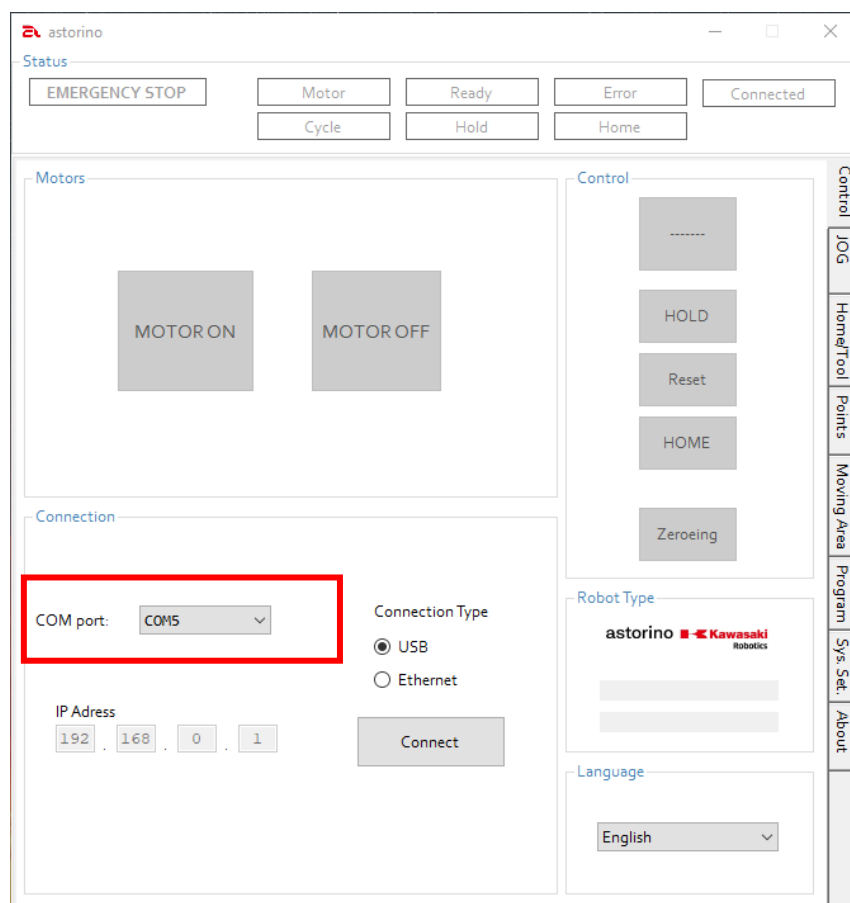


Rozpocznij instalację



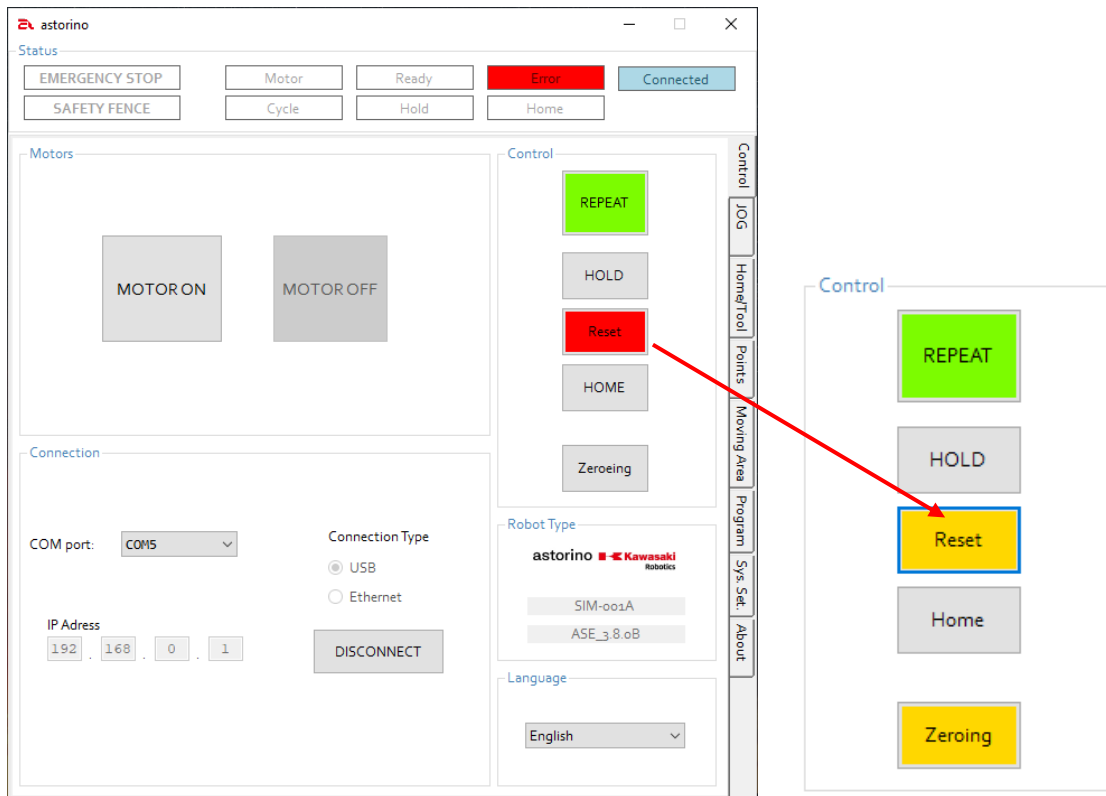
### 10.5 Przygotowanie Astorino do pracy

- Otwórz oprogramowanie Astorino.
- Port COM, do którego podłączony jest robot, powinien automatycznie pojawić się na liście rozwijanej w obszarze Połączenie w menu [Sterowanie].



## ASTORINO Instrukcja obsługi

- kliknij [Reset], gdy przycisk jest podświetlony na czerwono (sprawdź przycisk zatrzymania awaryjnego)

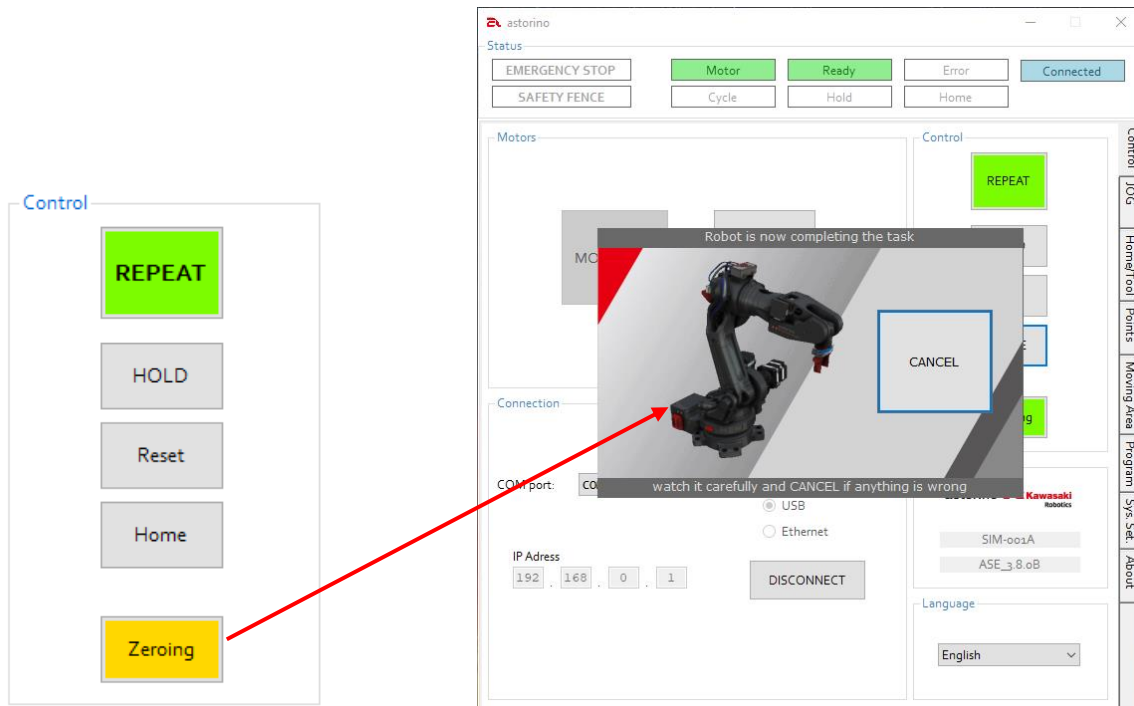


- Możesz teraz włączyć silniki, klikając szary przycisk [MOTOR ON]
- Kliknij żółte pole [Zerowanie], aby wykonać proces zerowania.

Zerowanie musi być wykonywane za każdym razem, gdy robot jest odłączany od zasilania lub silniki są wyłączone.

- Upewnij się, że robot w nic nie uderzy!

## ASTORINO Instrukcja obsługi



- Po zakończeniu zerowania robot stanie w pozycji 0 stopni na każdej osi. Teraz jest gotowy do programowania.



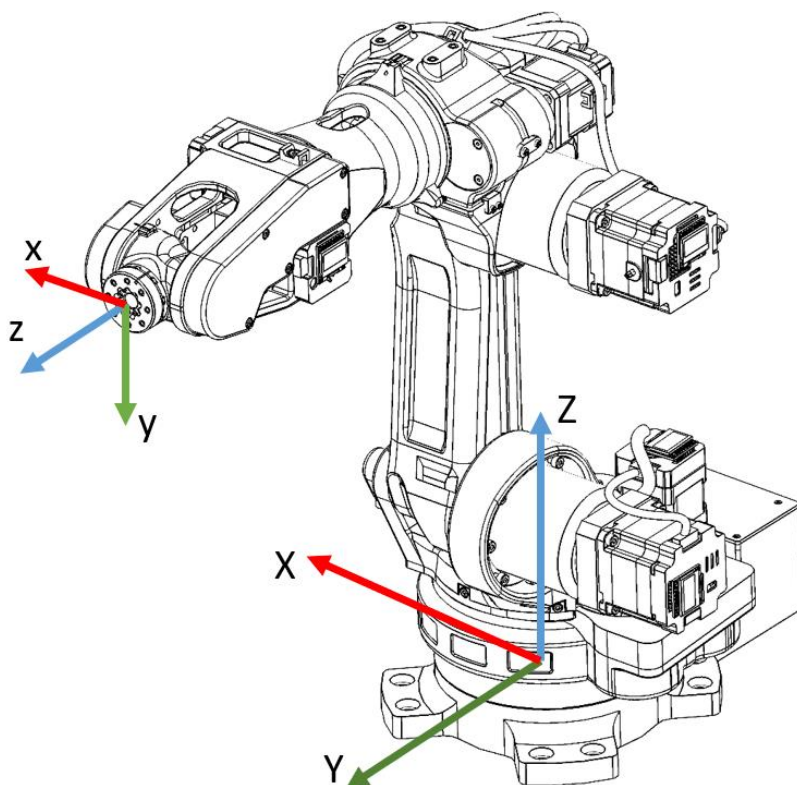
### UWAGA!

Pozycja pionowa jest standardową pozycją ramienia po zakończeniu procesu zerowania. Jeżeli ustawienia procedury zerowania zostały zmienione to pozycja końcowa może być inna!

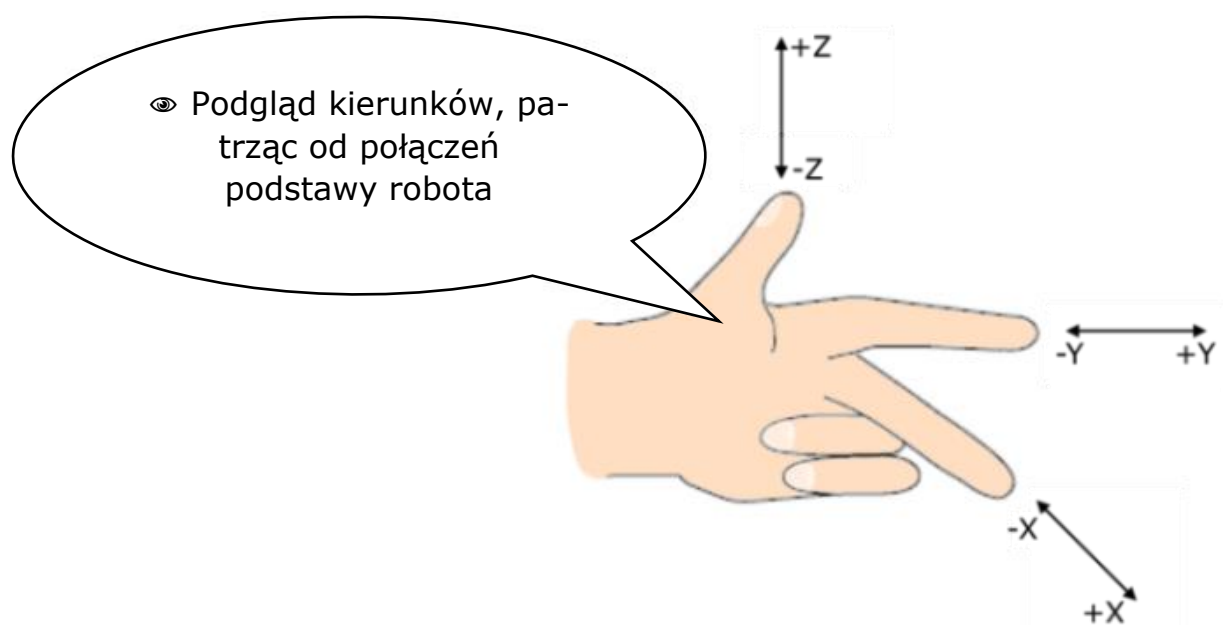


## 11 Układy współrzędnych

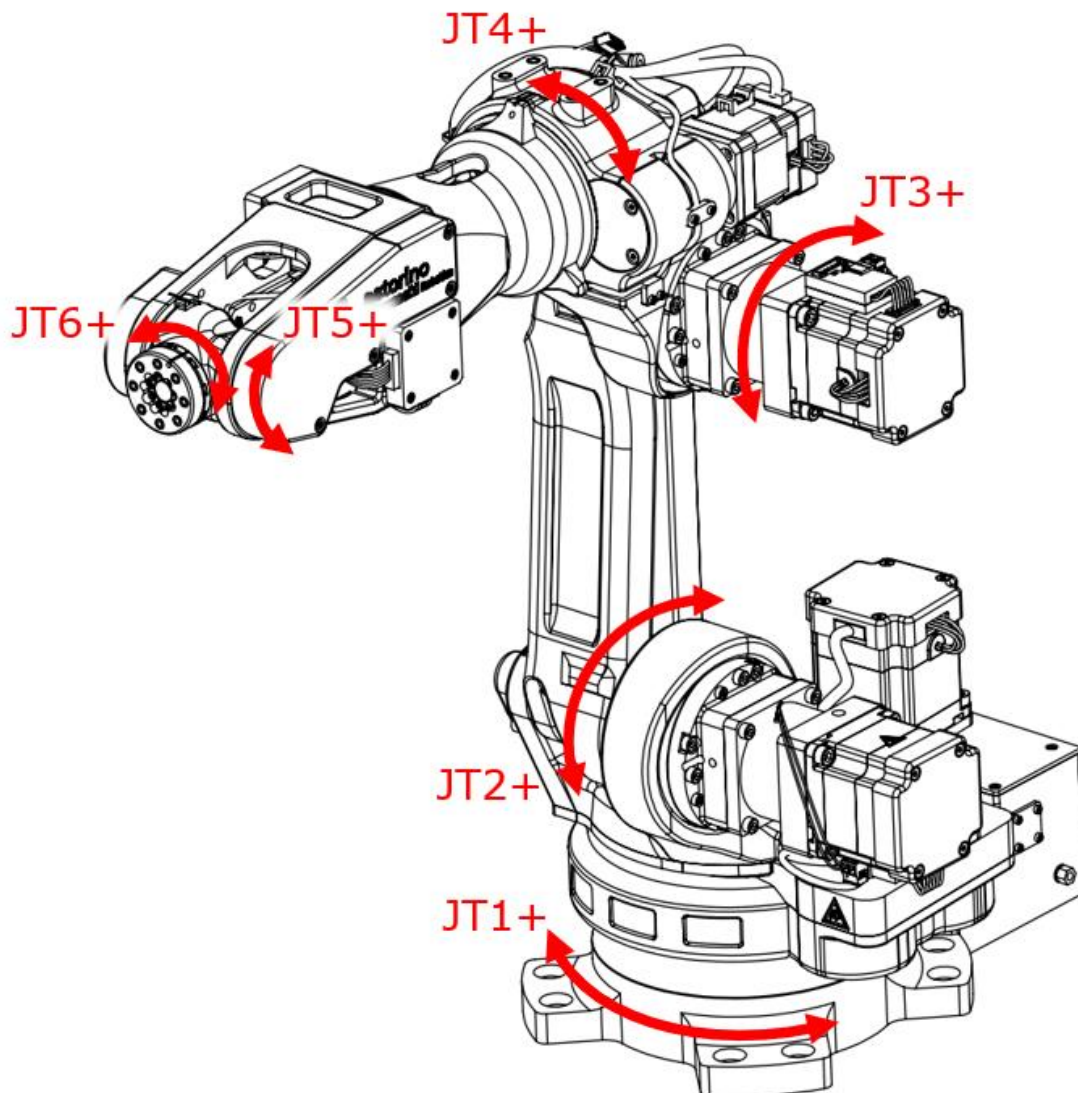
### 11.1 Układ współrzędnych BASE



Reguła lewej ręki pomaga zapamiętać kierunki osi:



## 11.2 Układ współrzędnych JOINT



**JT = oś**

Poszczególne połączenia są numerowane w porządku rosnącym, zaczynając od podstawy robota. JT oznacza oś.

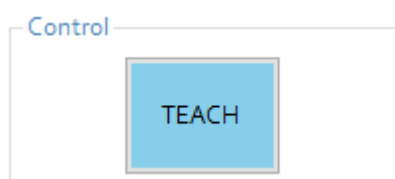
## 12 Tryby pracy robota

### 12.1 Tryb nauczania (Teach)

Tryb ten umożliwia ręczne operacje robota, takie jak poruszanie się, nauczanie. W tym trybie maksymalna prędkość jest ograniczona do 60 mm/s i 12 st./s dla każdej osi.

Nauczanie definiuje się jako programowanie robota do wykonywania wymaganych zadań i rejestrowanie danych pozycji.

Robot jest w trybie uczenia, gdy tło przełącznika [REPEAT/TEACH] jest niebieskie.

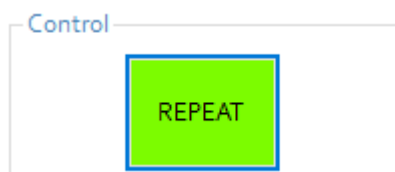


W trybie uczenia, gdy wejście Safety Fence jest stanie wysokim, operacje robota nie są ograniczone.

### 12.2 Tryb powtarzania (Repeat)

Odtwarzanie zawartość programu, który napisał użytkownik. W tym trybie prędkości nie są ograniczone, a robot może poruszać się maksymalnie z prędkością 250 mm / s.

Robot jest w trybie powtarzania, gdy tło przełącznika [REPEAT/TEACH] jest zielone.

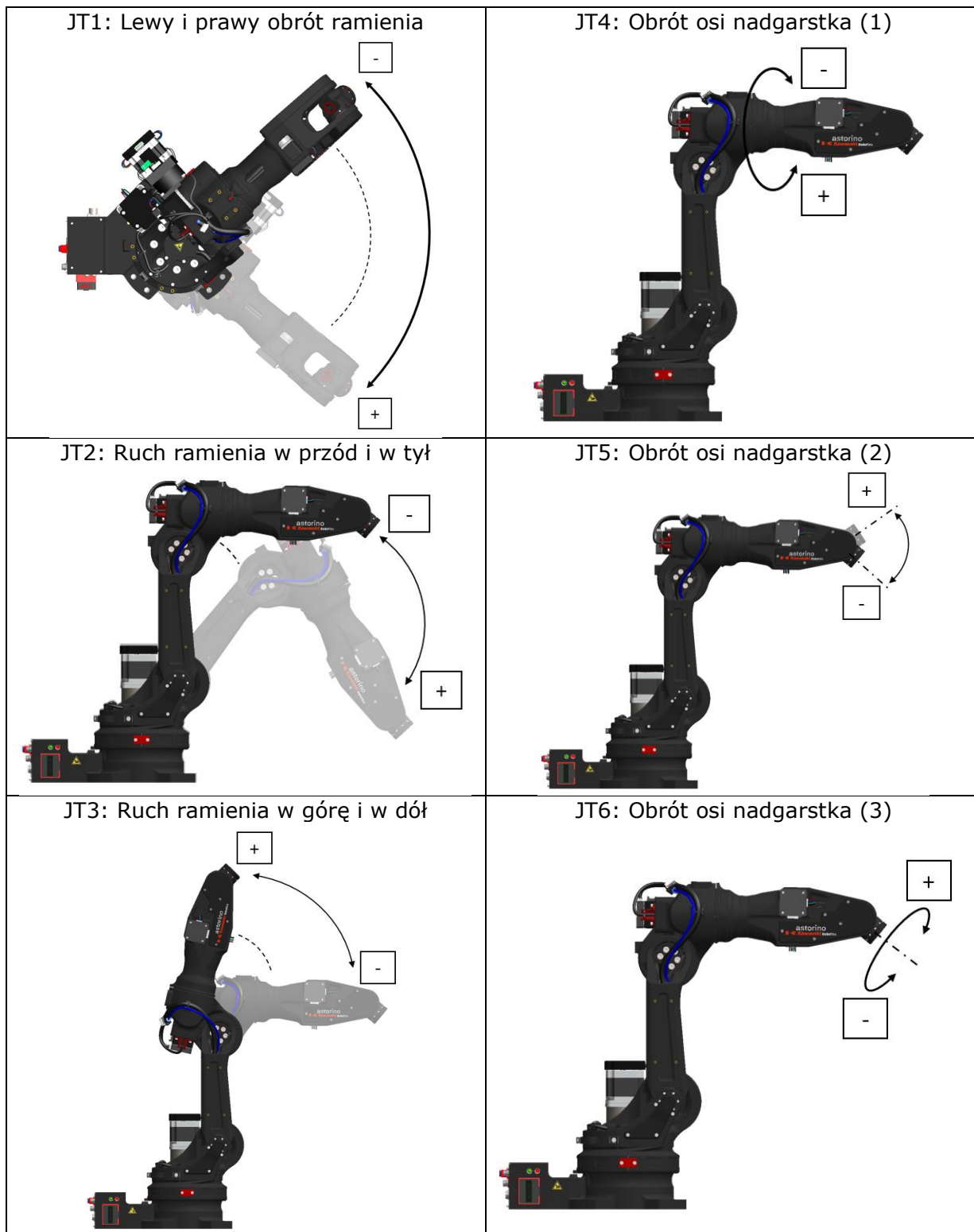


W trybie uczenia, gdy wejście Safety Fence jest stanie wysokim, operacje robota są ograniczone.

## 13 Ręczna obsługa robota

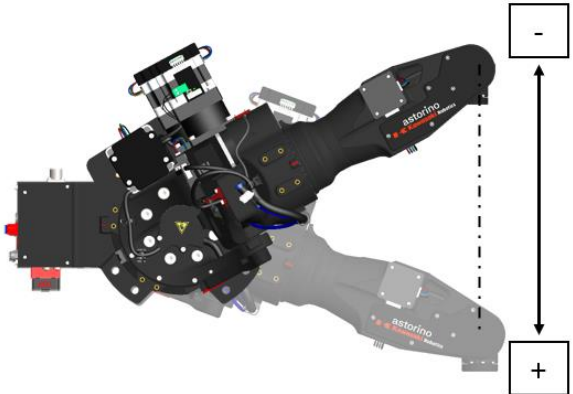
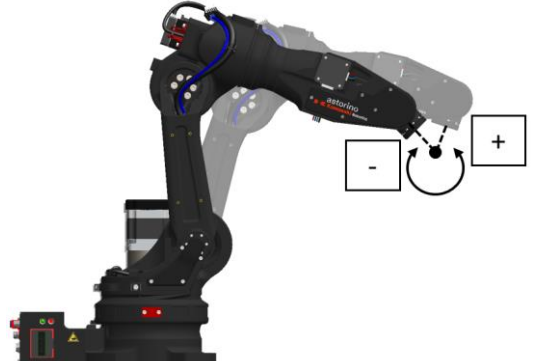
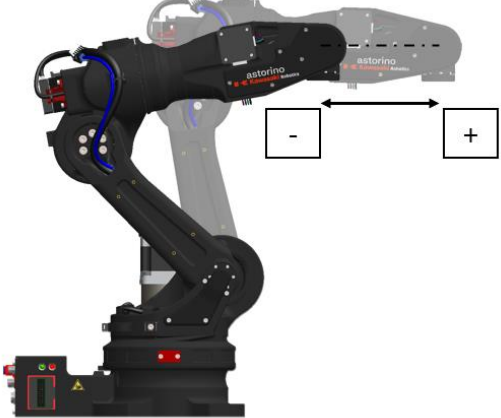
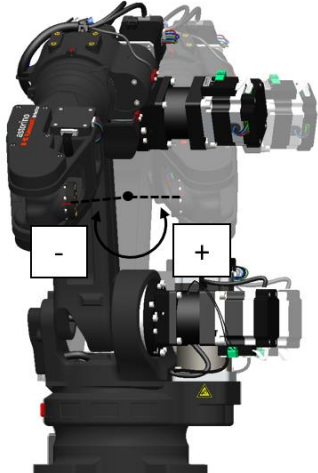
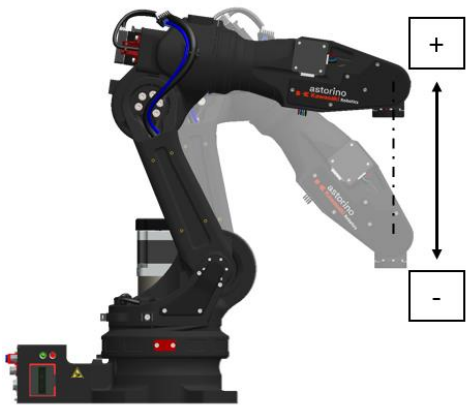
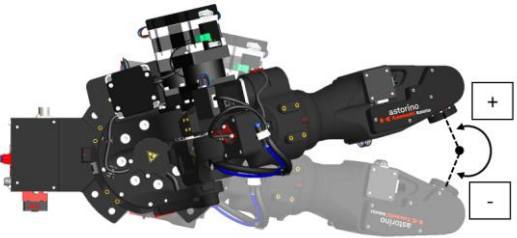
W trybie Teach opartym na aktualnie wybranym trybie ruchu (BASE, JOINT, TOOL) możliwe jest ręczne poruszanie ramieniem robota.

### 13.1 JOINT (Ruch złączowy)



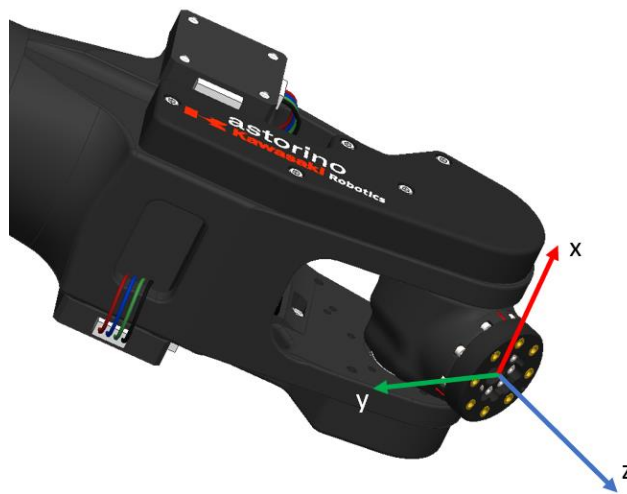
ASTORINO Instrukcja obsługi

**13.2 BASE (Ruch względem podstawowego układu wsp.)**

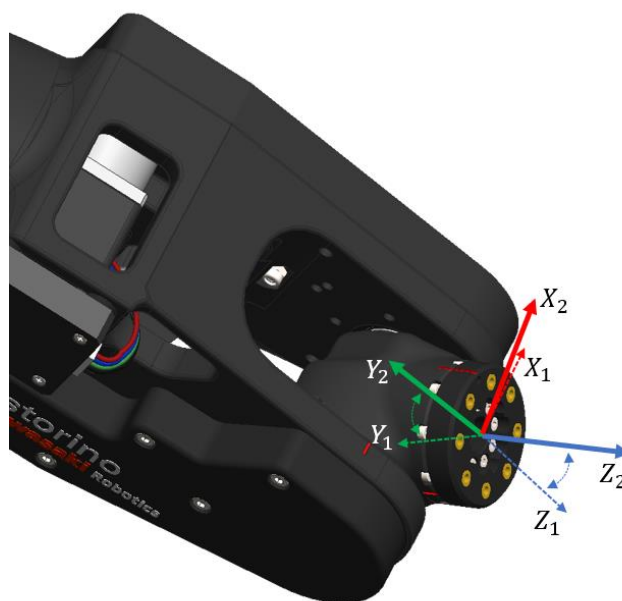
<p>X: Ruch równoległy do współrzędnej bazowej X (orientacja nadgarstka jest stała)</p> 	<p>RX: Obrót wokół współrzędnej X (TCP nie porusza się)</p> 
<p>Y: Ruch równoległy do współrzędnej bazowej Y (orientacja nadgarstka jest stała)</p> 	<p>RY: Obrót wokół współrzędnej Y (TCP nie porusza się)</p> 
<p>Z: Ruch równoległy do współrzędnej bazowej Z (orientacja nadgarstka jest stała)</p> 	<p>RZ: Obrót wokół współrzędnej Z (TCP nie porusza się)</p> 

### 13.3 TOOL (Ruch względem układu narzędzia)

Układ współrzędnych narzędzia jest zdefiniowany na narzędziu zainstalowanym na osi 6. Operacje oparte na tym układzie współrzędnych narzędzia będą różnić się kierunkiem ruchu w zależności od transformacji współrzędnych do współrzędnych narzędzia null (zerowy). Współrzędne narzędzia zmieniają się również wraz ze zmianą orientacji nadgarstka, jak pokazano na poniższych rysunkach, mimo że tylko przedramię porusza się bez poruszania osiami nadgarstka.

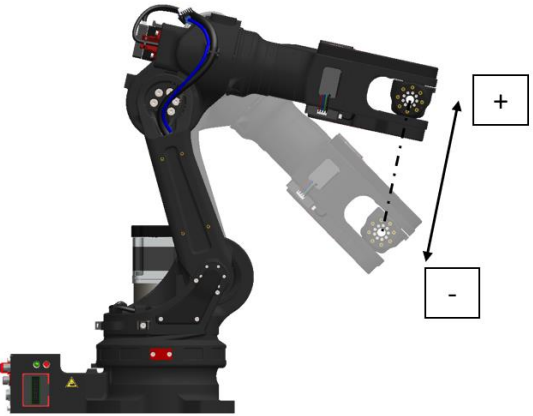
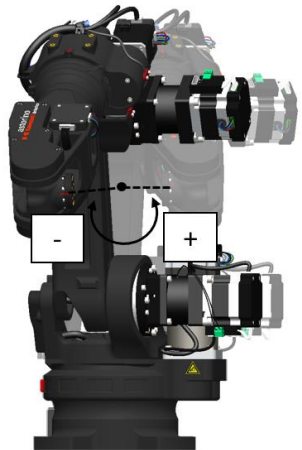
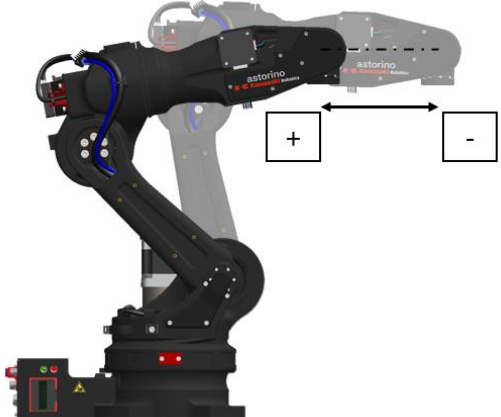
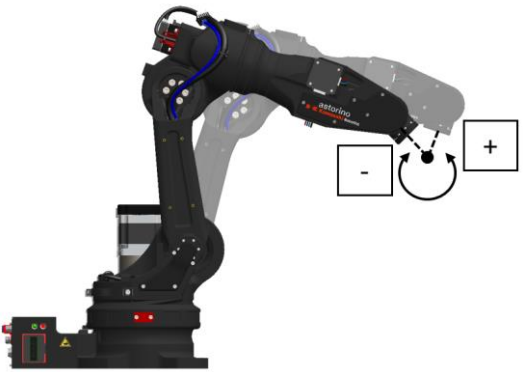
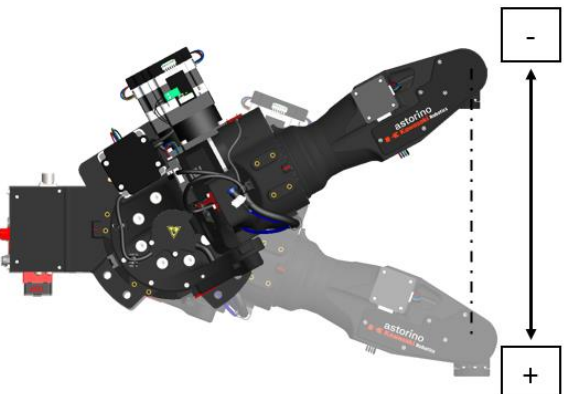
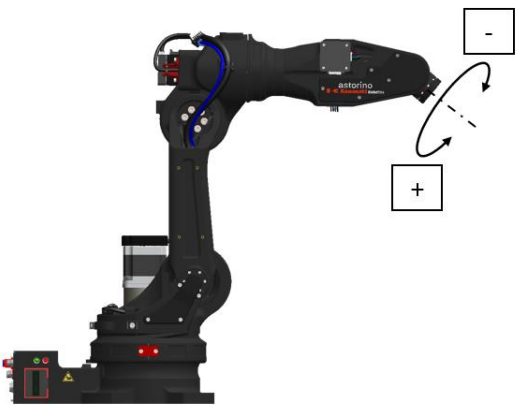


Ramię w innym miejscu i orientacji



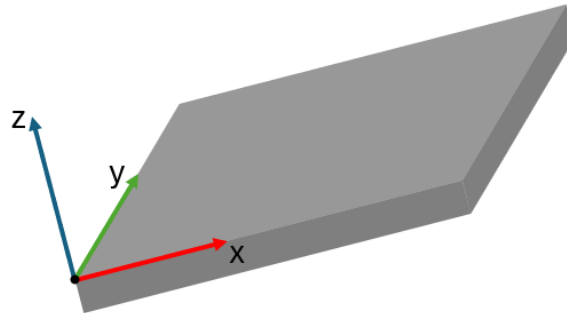


## ASTORINO Instrukcja obsługi

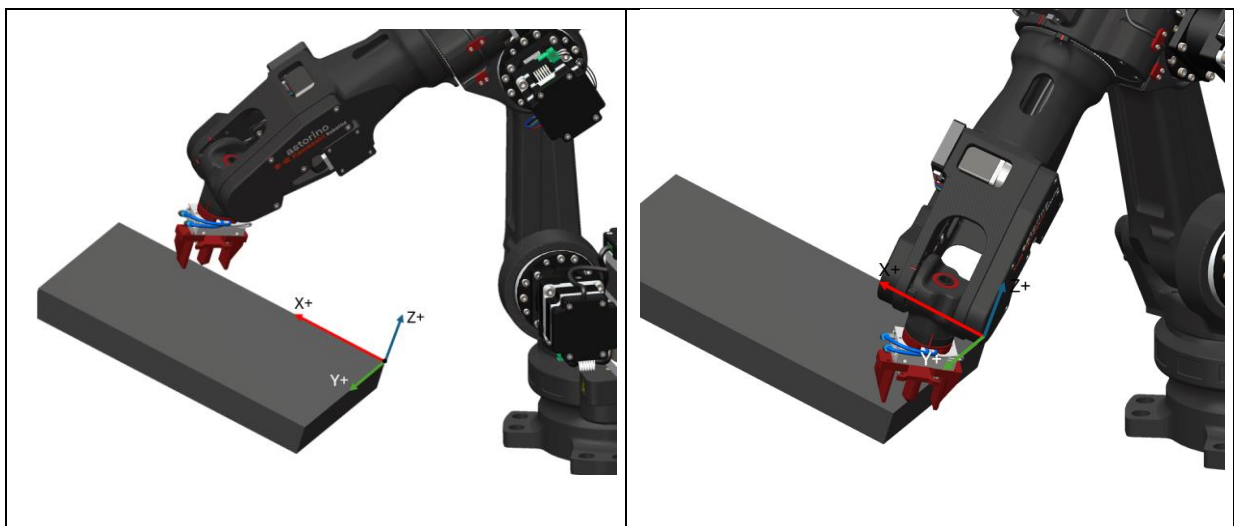
<p>x: Ruch równoległy do współrzędnej X narzędzia (orientacja nadgarstka jest stała)</p> 	<p>rx: Obrót wokół współrzędnej X narzędzia (TCP nie porusza się)</p> 
<p>y: Ruch równoległy do współrzędnej Y narzędzia (orientacja nadgarstka jest stała)</p> 	<p>ry: Obrót wokół współrzędnej Y narzędzia (TCP nie porusza się)</p> 
<p>z: Ruch równoległy do współrzędnej narzędzia Z (orientacja nadgarstka jest stała)</p> 	<p>rz: Obrót wokół współrzędnej narzędzia Z (TCP nie porusza się)</p> 

## 13.4 WORK

Układ współrzędnych WORK jest definiowany na dowolnej płaszczyźnie w przestrzeni, która znajduje się w zakresie ruchu robota. Operacje oparte na tym układzie współrzędnych będą się różnić kierunkiem ruchu w zależności od transformacji współrzędnych od układu BASE.



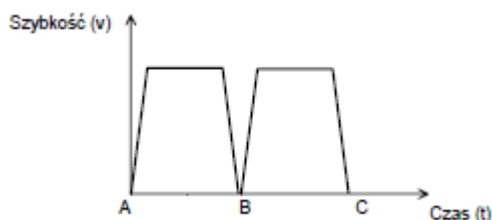
Ramię robota w różnych ułożeniach względem układu WORK





## 14 Ruch robota

Przyspieszenie dla drugiego segmentu rozpoczyna się po zakończeniu realizacji pierwszego segmentu, gdy bieżąca pozycja znajduje się w punkcie docelowym. Nachylenie wzrostu prędkości jest określane przez parametr ACCEL, a krawędź hamowania przez parametr DECEL.



Robot Astorino może poruszać się na trzy różne sposoby. Te sposoby nazywane są interpolacjami. Możemy wyróżnić:

- Interpolacja liniowa
- Interpolacja złączowa
- Interpolacja kołowa

W antropomorficznych ramionach robota (6 osi) istnieją pewne pozycje, które nazywane są osobliwościami. Pojedyncza pozycja, w której może wystąpić problem strukturalnego niekontrolowanego położenia, istnieje, gdy na przykład JT4 i JT6 są równoległe do siebie lub JT1 i JT6 są równoległe do siebie. Konfiguracje te zwracają wiele matematycznych rozwiązań zadania kinematyki odwrotnej, a zatem ruch przez te punkty może być nieprzewidywalny i wprowadzać wiele bardzo szybkich ruchów osi.

Przykłady pozycji osobliwych

JT4 i JT6 są równoległe

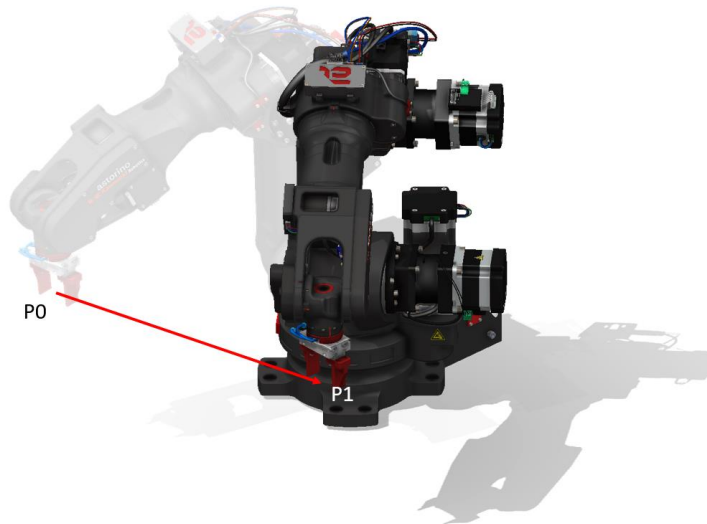


JT1 i JT6 są równoległe



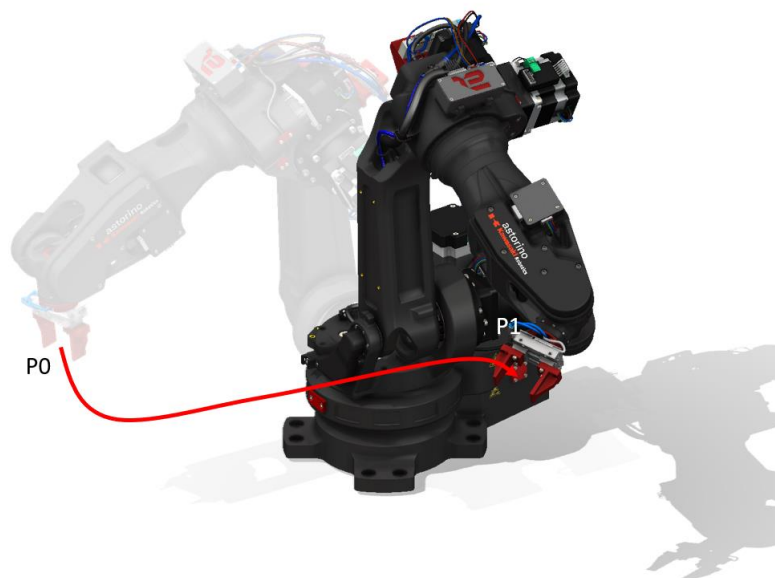
## 14.1 Interpolacja liniowa

W tego typu interpolacji robot porusza się z bieżącej pozycji do miejsca docelowego w taki sposób, że TCP porusza się po linii prostej w przestrzeni 3D.



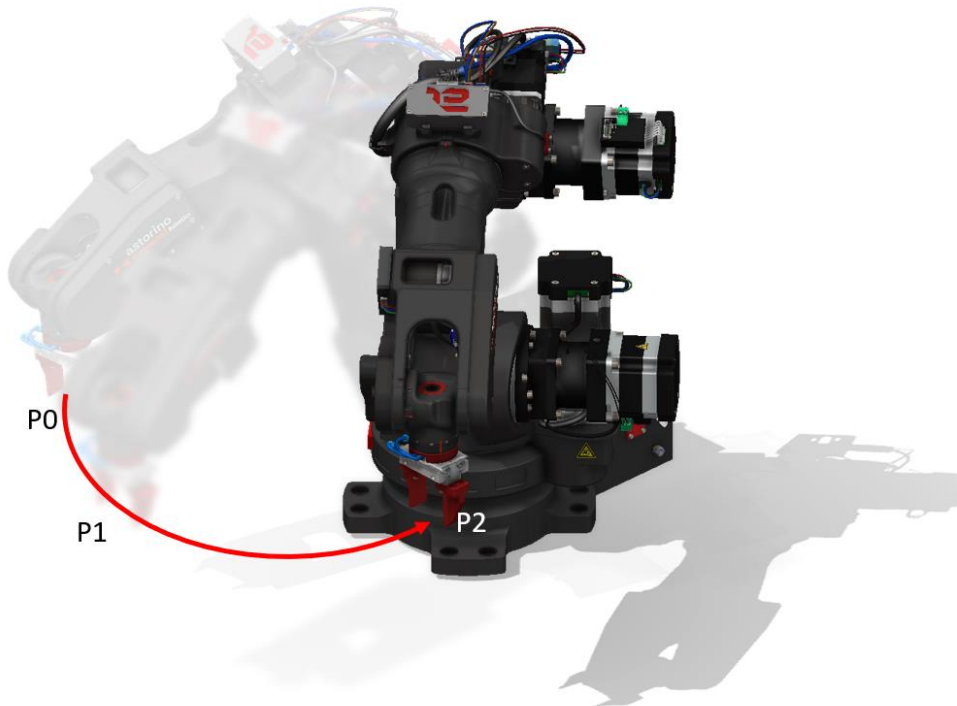
## 14.2 Interpolacja złączowa

W tym typie interpolacji robot porusza się z bieżącej pozycji do miejsca docelowego w taki sposób, że wszystkie osie kończą ruch w tym samym czasie. Ten ruch tworzy nieprzewidywalną ścieżkę TCP w przestrzeni 3D. Ruch ten pozwalana uniknąć problemu ruchu przez punkty osobliwe.



### 14.3 Interpolacja kołowa

W tym typie ruchu robot porusza się z bieżącej pozycji do miejsca docelowego przez środkowy punkt w taki sposób, że TCP tworzy łuk w przestrzeni 3D.

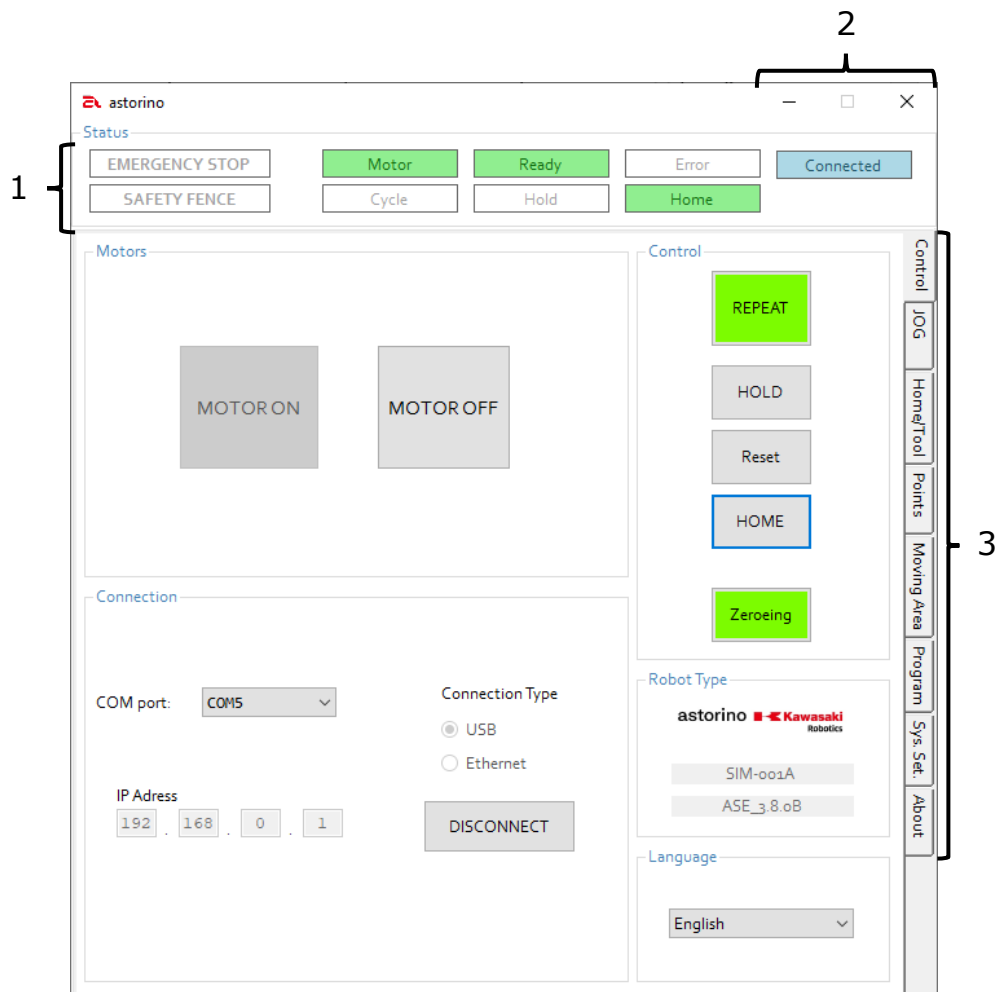


## 15 Oprogramowanie Astorino

### 15.1 Podstawowe informacje

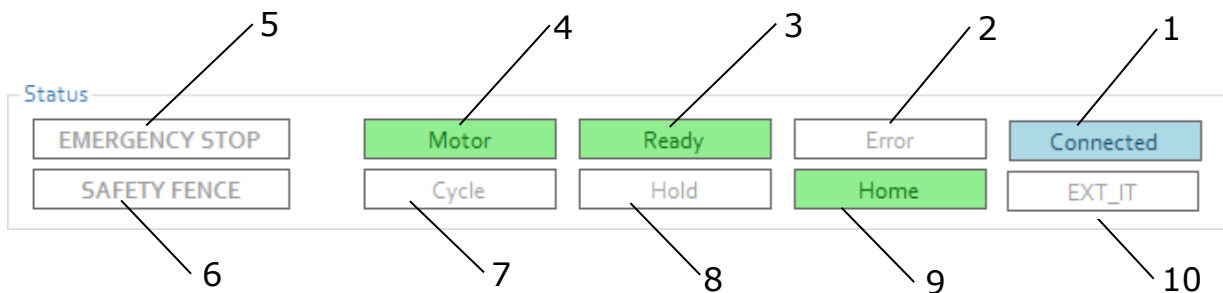
Wszystkie dane są przechowywane na wewnętrznej karcie micro SD, która znajduje się na płycie mikrokontrolera wewnątrz podstawy robota. Jeśli robot jest wyłączony, dane użytkownika nie są usuwane.

Przegląd okna głównego.



- |                       |  |
|-----------------------|--|
| 1. Status             | Aktualny status podłączonego robota              |
| 2. Kontrola aplikacji | Zamykanie lub minimalizowanie aplikacji astorino |
| 3. Karty operacji     | Przełączanie między różnymi kartami operacji     |

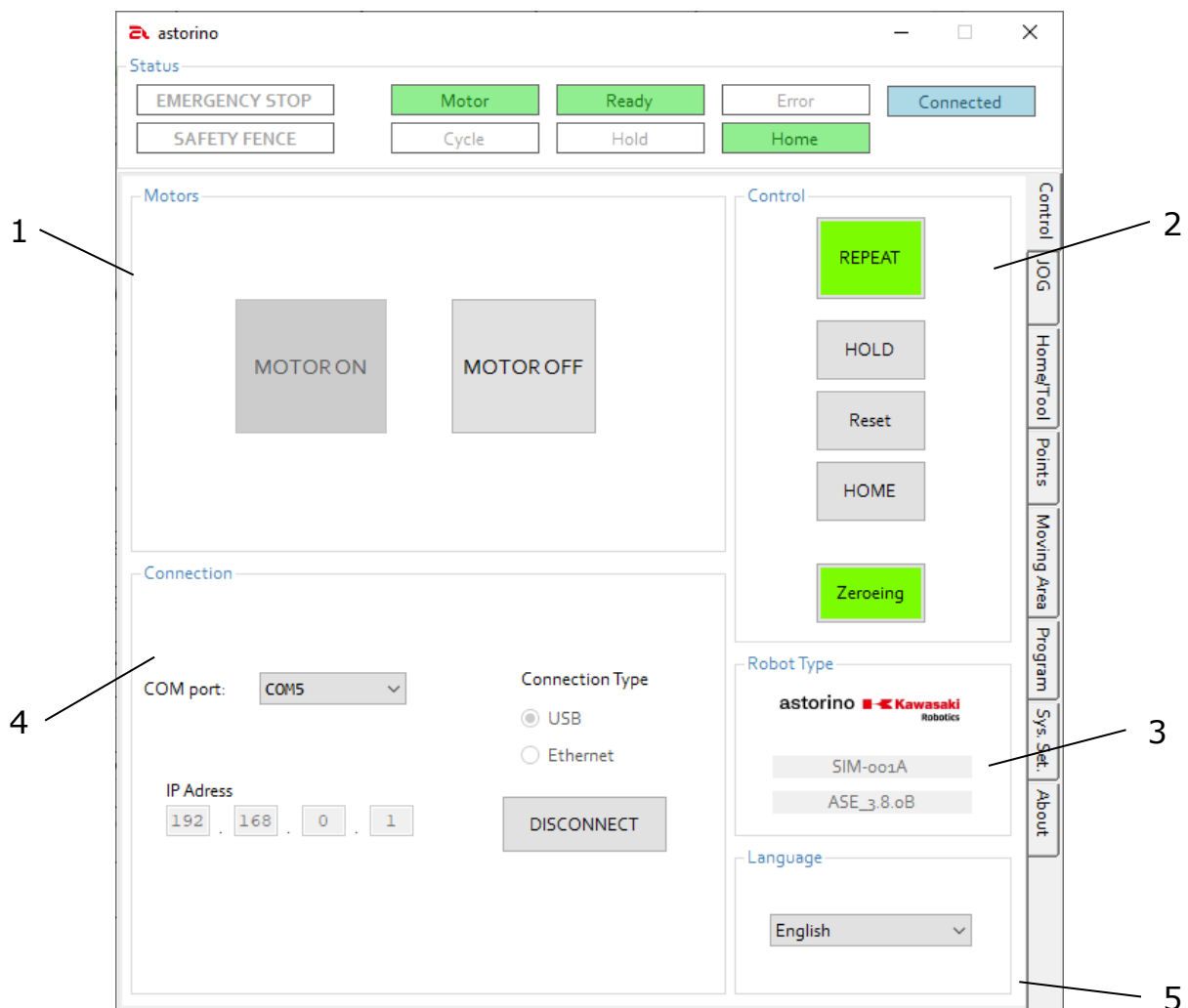
## 15.2 Obszar Status



Gdy tło danego pola podświetli się oznacza to, że:

1. Connected	Podłączony do oprogramowania Astorino
2. Error	Wystąpił błąd
3. Ready	Jeśli nie ma zatrzymania awaryjnego i nie występują żadne błędy, napędy są załączone i zerowanie zostało zakończone
4. Motors	Napędy są załączone
5. EMERGENCY STOP	Wyłącznik awaryjny jest wciśnięty i aktywny
6. SAFETY FENCE	Zatrzymanie ogrodzenia ochronnego jest aktywne
7. Cycle	gdy sekwencja programu jest aktywowana
8. Hold	kiedy robot został zatrzymany
9. Home	robot znajduje się w pozycji domowej
10. EXT_IT	gdy robot został zatrzymany przez zewnętrzny sygnał

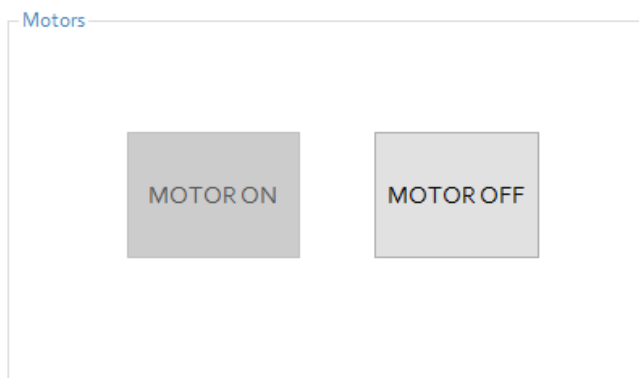
## 15.3 Karta Control / Sterowanie



MOTORS	Stan silników i kontrola nad silnikami
CONTROL	Sterowanie trybem pracy, zatrzymanie robota lub przejście do pozycji początkowej, potwierdzenie błędu i zerowanie
ROBOT-TYPE	Wersja oprogramowania sprzętowego robota i numer seryjny
Connection	Wybór połączenia i konfiguracja interfejsu, ustanowienie połączenia lub rozłączenie
Language	Wybór języka wyświetlanego języka

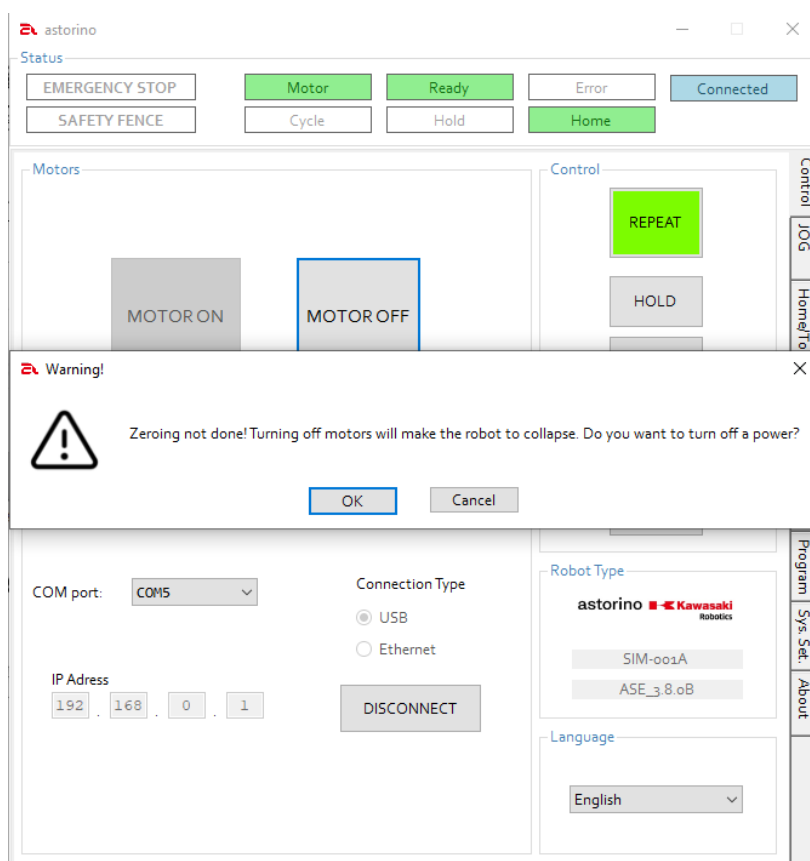
## ASTORINO Instrukcja obsługi

### 15.3.1 Obszar Motors / Napędy



Kliknij [MOTOR ON], aby aktywować sterowniki silników krokowych. Jest to możliwe tylko wtedy, gdy nie występuje błąd! (Pole  błędu)

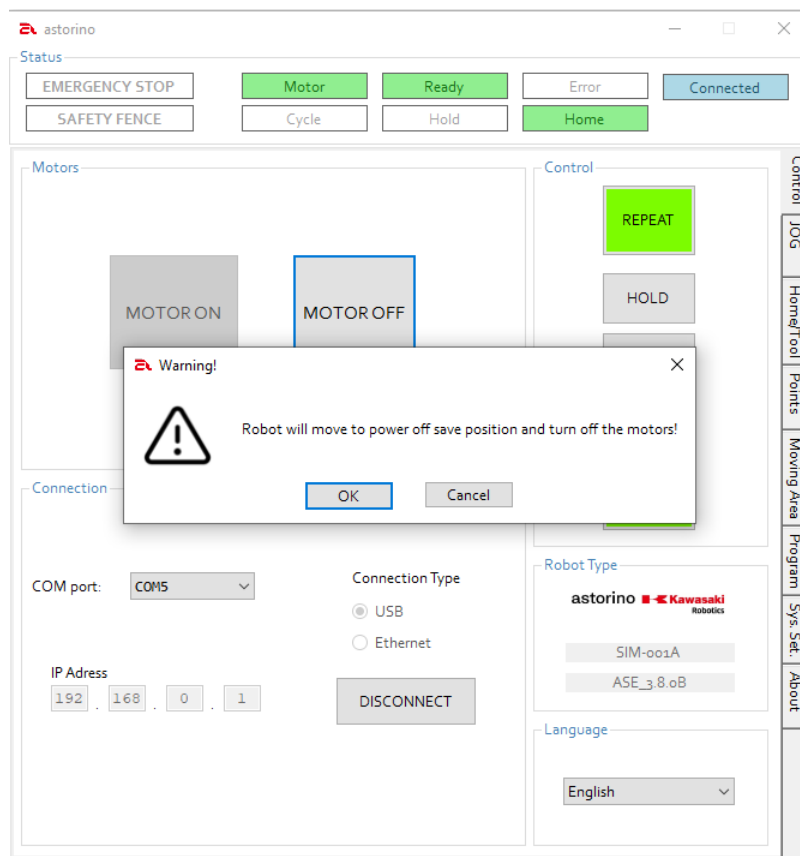
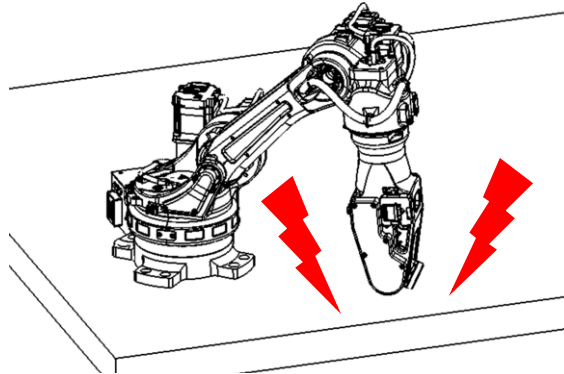
Jeśli silniki są gotowe, jest to wyświetlane w obszarze **Status** ( Motors ).



Kliknij **MOTOR OFF** Aby wyłączyć sterowniki silnika krokowego.

 **OSTRZEŻENIE**

**Jeśli nie wykonano zerowania osi, zachowaj szczególną ostrożność i przytrzymaj oś 4, ponieważ robot może się upaść.**



Jeśli [Zerowanie] jest zakończone, Astorino automatycznie uda się do bezpiecznej pozycji wyłączenia po potwierdzeniu komunikatu ostrzegawczego.

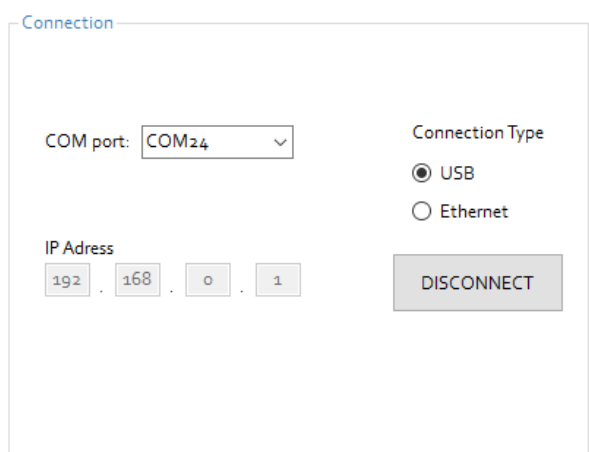


### 15.3.2 Obszar Control / Sterowanie



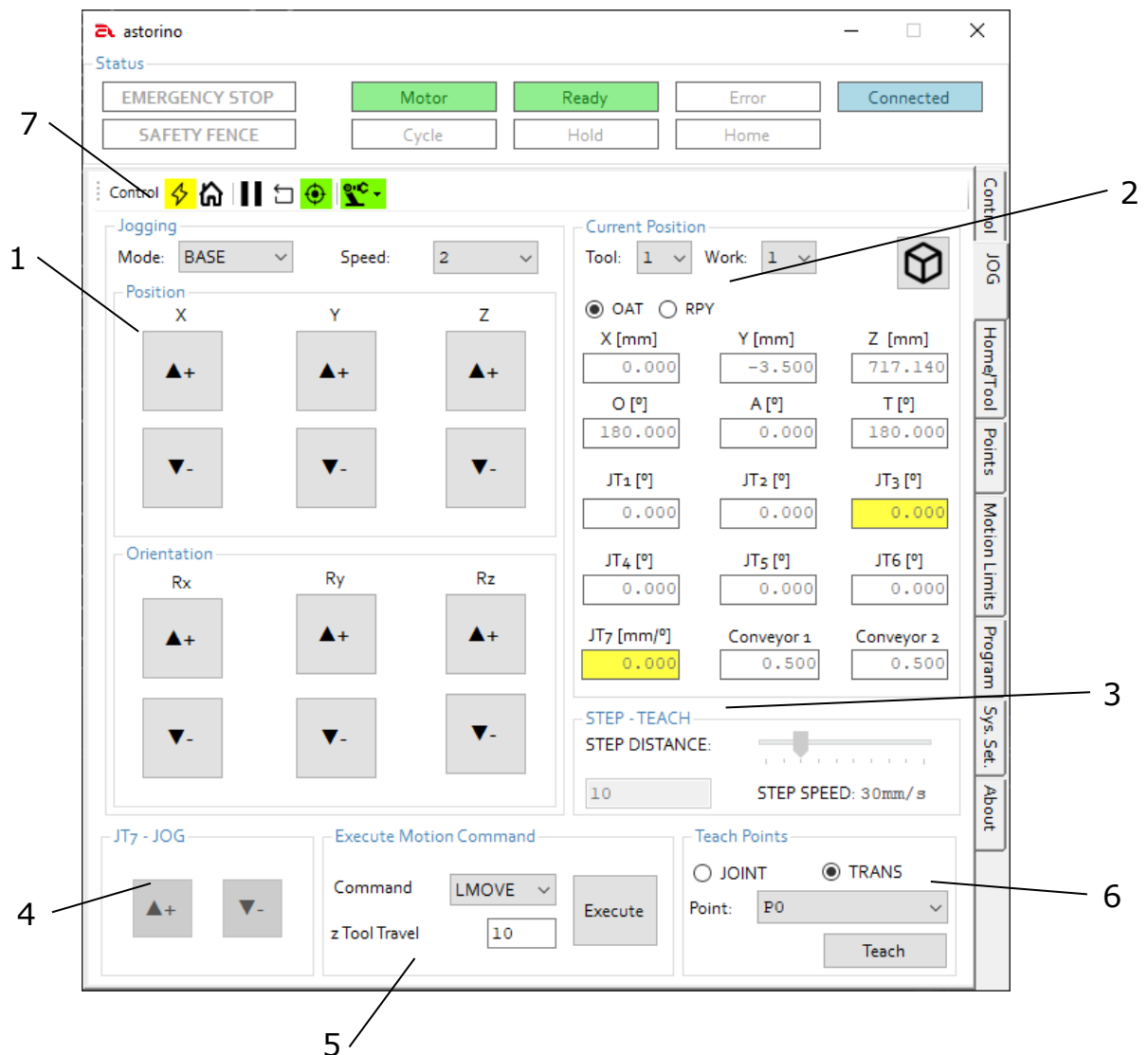
1. **[REPEAT/TEACH]** – kliknij ten przycisk, aby zmienić tryb pracy. Przycisk zmienia kolor na zielony, gdy jest w trybie automatycznym (REPEAT) i niebieski, gdy jest w trybie nauczania (TEACH).
2. **[HOLD]** – kliknij, aby zatrzymać robota.
3. **[Reset]** – Kliknij, aby zresetować błędy.
4. **[Home]** – kliknij, aby robot wykonał ruch do pozycji domowej.
5. **[Zerowanie]** – kliknij, aby wyzerować osie robota (np. po każdym restarcie).

### 15.3.3 Obszar Connection / Połączenie



1. Port COM - wyświetla porty COM, do których podłączony jest robot.
2. Connection Type - wybierz typ używanego połączenia (USB/ Ethernet).
3. IP Adress - wprowadź adres IP robota.
4. **[CONNECT/DISCONNECT]** – Kliknij przycisk, aby połączyć się z robotem lub go odłączyć.

## 15.4 Karta JOG



1. Jogging

Wybierz tryb ruchu i prędkość

2. Current Position

wybór narzędzia (Tool), wyświetlanie kątów\*, aktualna pozycja robota

3. STEP-TEACH

ustaw rozmiar kroku i prędkość kroku

4. JT7 – JOG

przesuń oś liniową (JT7) - jeśli jest dostępna

5. Execute Motion Command

Przemieszczenie do zaznaczonego punktu z sekcji Teach Points

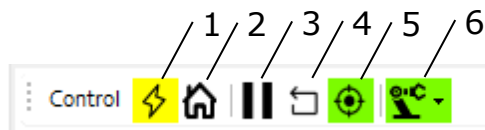
6. Teach Point

Wybierz punkt, do którego chcesz się uczyć lub do którego chcesz się przenieść

## ASTORINO Instrukcja obsługi

### 7. Control

Zduplikowana funkcjonalność z karty Sterowanie



1. WŁĄCZANIE/WYŁĄCZANIE SILNIKA
2. POZYCJA DOMOWA
3. WSTRZYMANIE PRACY
4. RESET
5. ZEROWANIE
6. PRZEŁĄCZANIE REPEAT/TEACH

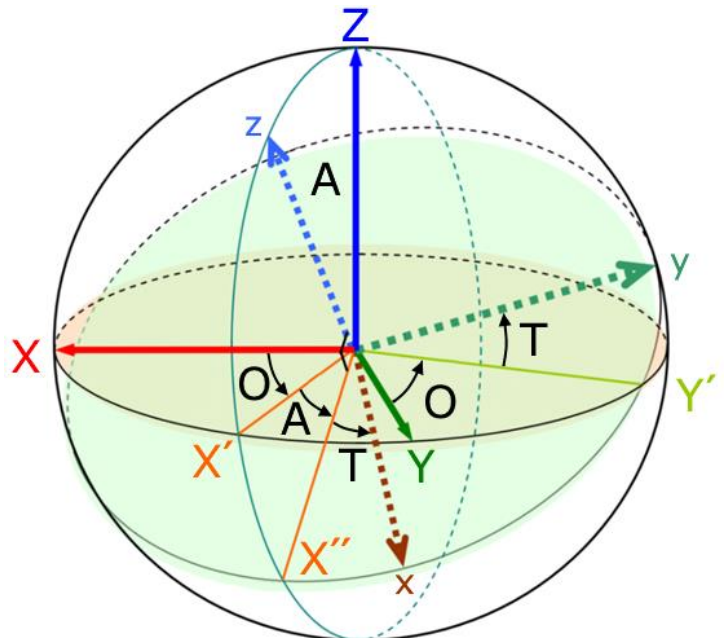
\* Wyświetlanie kąta lub zastosowane sekwencje obrotu

Podczas obliczania ruchu robot wykorzystuje kąty EULERA (OAT) do obliczania ścieżki i pozycji robota.

Dla ułatwienia nauczania **Roll-Pitch-Yaw** jest używany, ponieważ jest bardziej intuicyjny dla użytkownika, a następnie automatycznie konwertowany do pozycji OAT przez robota.

### Klasyczne kąty Eulera O,A,T

Format pozycji (POSE) używany przez roboty Kawasaki składa się z pozycji XYZ w milimetrach i orientacji OAT, która jest określona przez trzy kąty w stopniach, gdzie **<O>** obraca się wokół osi **Z**, **<A>** obraca się wokół obróconej osi **Y (Y')**, a **<T>** obraca się wokół obróconej osi **Z**.



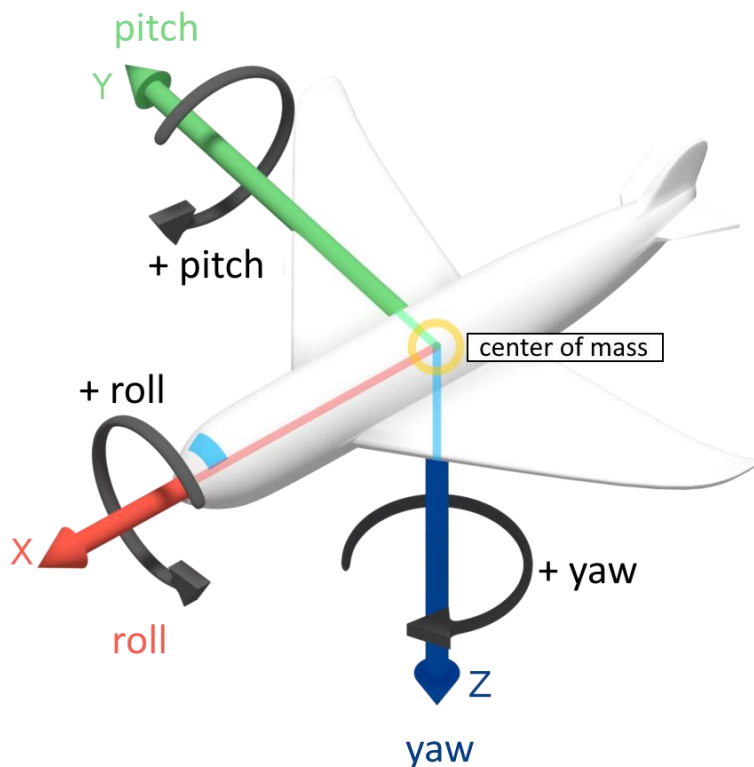
### Roll-Pitch-Yaw (RPY):

Kąty Roll-Pitch-Yaw to specjalne kąty Eulera (kąty położenia), które są używane do opisu orientacji obiektu w przestrzeni 3-wymiarowej.

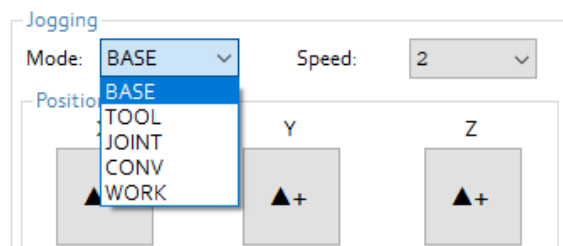
**Roll, oś obrotu X**

**Pitch, oś obrotu Y**

**Yaw, oś obrotu Z**

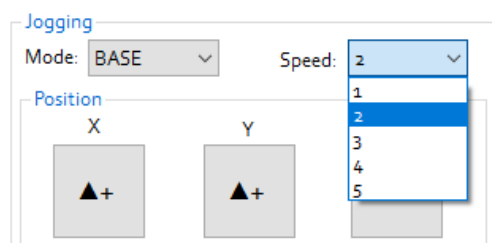


### 15.4.1 Obszar Jogging



#### Wybierz system odniesienia:

- BASE (współrzędne bazowe)
- TOOL (współrzędne narzędzia)
- JOINT (tryb jednoosiowy)
- CONV (synchroniczny w połączeniu z opcjonalnym enkoderm zewnętrznym - przenośnikiem)
- WORK (współrzędne układu WORK)

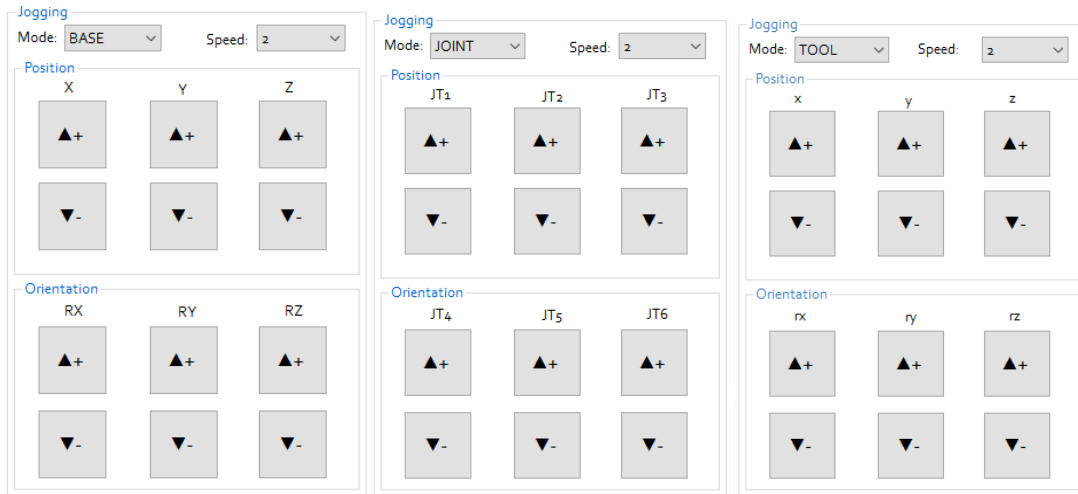


Ustawianie prędkości ruchu robota.

Szybkość	Tryb artezyjski C	Tryb jednoosiowy (JOINT)
1	Ruch na odległość	Obrót o stały kąt
2	5 mm/s	2°/s
3	10 mm/s	4°/s
4	30 mm/s	8°/s
5	60 mm/s	12°/s

## ASTORINO Instrukcja obsługi

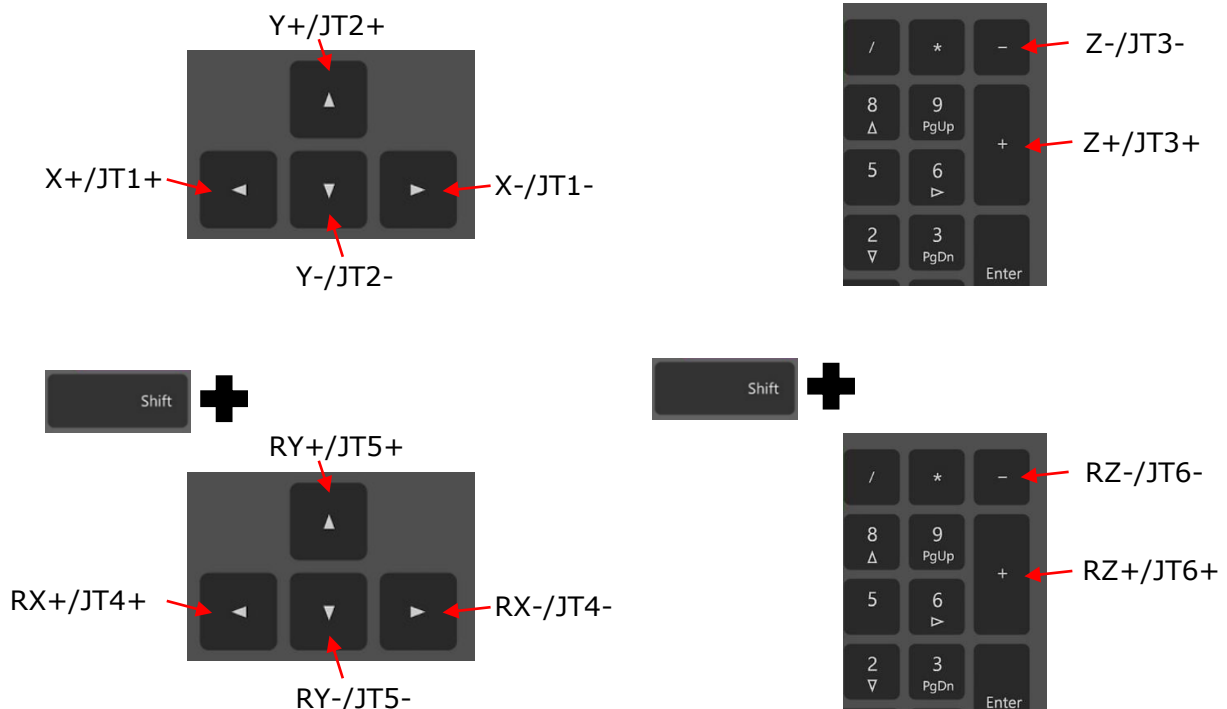
"▲+" i "▼-" poruszają robota w trybie uczenia z wybraną prędkością:



- BASE (współrzędne kartezjańskie): X, Y, Z, RX, RY, RZ
- TOOL (współrzędne kartezjańskie): x, y, z, rx, ry, rz
- JOINT (pojedyncza oś): JT1, JT2, JT3, JT4, JT5, JT6
- CONV (współrzędne kartezjańskie): X, Y, Z, RX, RY, RZ
- WORK (współrzędne kartezjańskie): X, Y, Z, RX, RY, RZ

Jeśli aktualnie wybrana prędkość wynosi 1 (krok), to po naciśnięciu przycisków JOG (+/-) robot wykonuje ruch krokowy. Ruch krokowy jest ustawiony w sekcji STEP-TEACH.

Dodatkowo robotem można poruszać za pomocą klawiatury:



## 15.4.2 Obszar Current Position / Obecna Pozycja

Current Position

Tool: 1 Work: 1

OAT  RPY

X [mm]	Y [mm]	Z [mm]
0.000	-3.500	717.140
O [°]	A [°]	T [°]
180.000	0.000	180.000
JT1 [°]	JT2 [°]	JT3 [°]
0.000	0.000	0.000
JT4 [°]	JT5 [°]	JT6 [°]
0.000	0.000	0.000
JT7 [mm/°]	Conveyor 1	Conveyor 2
0.000	0.500	0.500

### Wybór narzędzia

(1-3, 4 – rezerwacja na układ programowy)

pokaż wizualizację

Załączony tryb DryRun (praca robot bez fizycznego ruchu)

Wybór układu WORK (1-2)

przełączanie pomiędzy wyświetlaniem kątów OAT lub RPY

Bieżąca pozycja ramienia

## 15.4.3 Obszar Step - Teach / Ustawienia kroku

STEP - TEACH

STEP DISTANCE: 10

STEP SPEED: 3deg/s

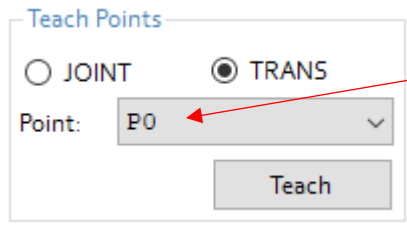
### ODLEGŁOŚĆ KROKU

wartość w mm lub ° dla ruchu

### PRĘDKOŚĆ KROKU

Prędkość w % lub mm/s

### 15.4.4 Obszar Teach Point / Naucz punkt

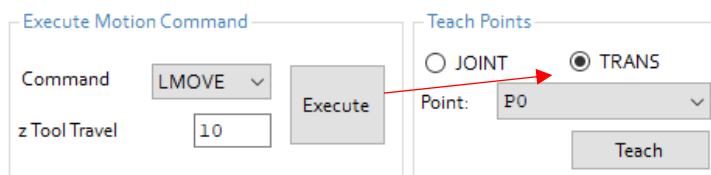


Wybierz punkt z listy, który ma zostać zapisany.

Wybierz, czy do punktu ma być zapisany w wartościach kartezjańskich (TRANS), czy kątów osi (JOINT).

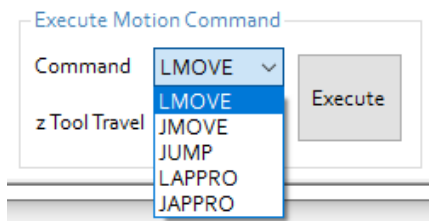
Na koniec kliknij przycisk [Teach], aby zapisać ten punkt w pamięci robota.

### 15.4.5 Obszar Execute Motion Command / Wykonaj Polecenie Ruchu

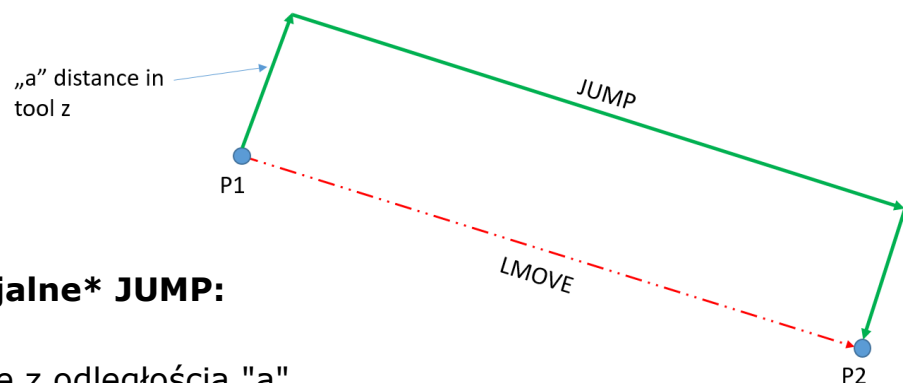


Command (możliwe polecenia ruchu):

- LMOVE (liniowy)
- JMOVE (złączowy)
- JUMP (specjalny\*)
- LAPPRO
- JAPPRO •



Po kliknięciu przycisku [Execute] polecenie jest wykonywane dla punktu wybranego w obszarze Teach Point.



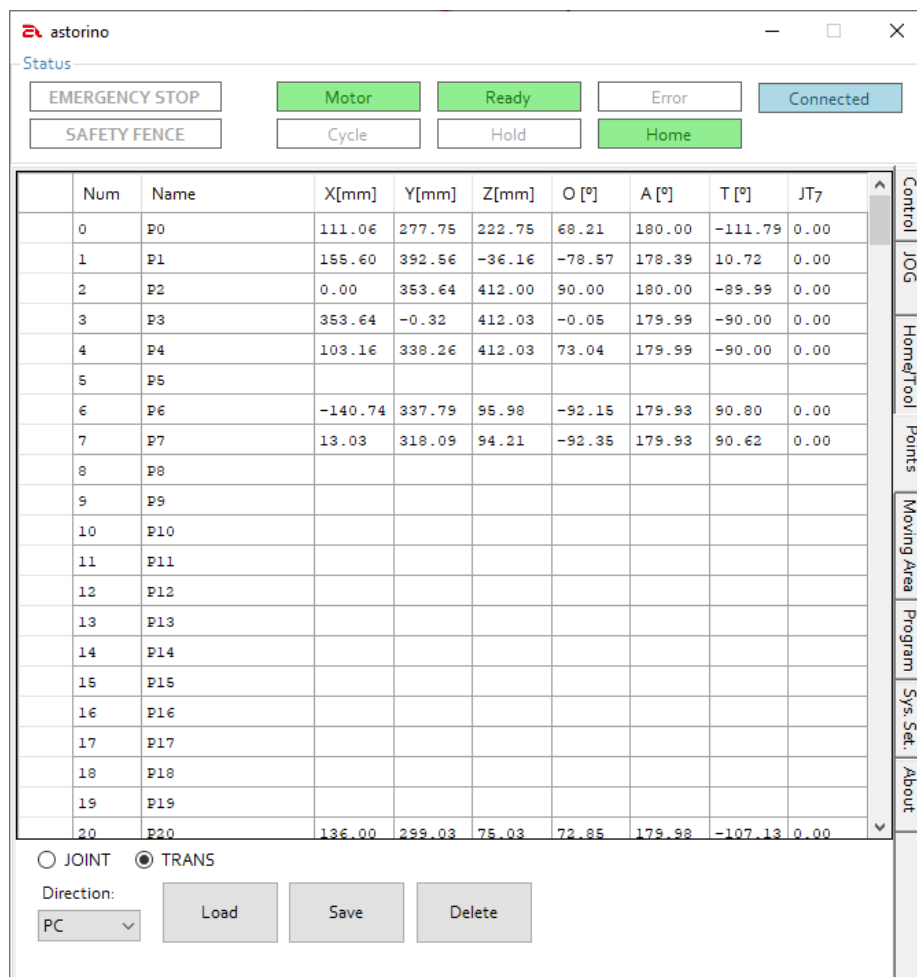
#### Polecenie specjalne\* JUMP:

Robot porusza się z odległością "a" do zdefiniowania od jednego punktu (np. P1) do następnego (np. P2).

Odległość "a" jest kierunkiem Z we współrzędnych narzędzia



## 15.5 Karta Points / Punkty



Num	Name	X[mm]	Y[mm]	Z[mm]	O [°]	A [°]	T [°]	JT7
0	P0	111.06	277.75	222.75	68.21	180.00	-111.79	0.00
1	P1	155.60	392.56	-36.16	-78.57	178.39	10.72	0.00
2	P2	0.00	353.64	412.00	90.00	180.00	-89.99	0.00
3	P3	353.64	-0.32	412.03	-0.05	179.99	-90.00	0.00
4	P4	103.16	338.26	412.03	73.04	179.99	-90.00	0.00
5	P5							
6	P6	-140.74	337.79	95.98	-92.15	179.93	90.80	0.00
7	P7	13.03	318.09	94.21	-92.35	179.93	90.62	0.00
8	P8							
9	P9							
10	P10							
11	P11							
12	P12							
13	P13							
14	P14							
15	P15							
16	P16							
17	P17							
18	P18							
19	P19							
20	P20	136.00	299.03	75.03	72.85	179.98	-107.13	0.00

W tej zakładce wszystkie punkty zapisane w oprogramowaniu są wyświetlane w formie tabelarycznej.

Można wyświetlić wszystkie punkty TRANS lub JOINT. Punkty od 0 do 99 są oznaczone jako Px na przykład P0 lub P10, punkty od 100 do 255 są punktami użytkownika i mają nazwę określoną przez użytkownika.

Z listy rozwijanej można wybrać kierunek, w którym mają być ładowane lub zapisywane dane punktu. Możesz wybrać podłączony komputer lub sterownik robota.

Przycisk **[LOAD]** ładuje dane z pamięci robota lub z plików \*.loc do sterownika robota ASTORINO.

Za pomocą przycisku **[ZAPISZ]** dane w pamięci robota lub w pliku \*.loc w pliku \*.loc na komputerze.

Przycisk **[USUŃ]** usuwa wybrany element z oprogramowania Astorino i z pamięci robota.

## 15.6 Karta Home/Tool / Pozycja domowa/Tool

### Obszar Pozycja domowa/Home

Przycisk [Set Home] służy do zapisywania aktualnej pozycji robota jako pozycji domowej.

Kliknij pole  wyboru Manual, aby ręcznie wprowadzić dane pozycji HOME. Następnie kliknij przycisk [Set Home], aby zapisać dane w pamięci robota.

Naciśnięcie przycisku [Home] na zakładce Control powoduje, że robot powróci w przyszłości do zapisanej pozycji!

## 15.6.1 Obszar Tool

**Tool**

Tool: 1  OAT  RPY

Tool X [mm]:

Tool Y [mm]:

Tool Z [mm]:

Tool Rx [°]:

Tool Ry [°]:

Tool Rz [°]:

Control

JOG

Points

Home/Tool

Moving Area

Programs

Sys

Różne narzędzia, takie jak chwytaki, końcówki lub inne, można wywołać i sparametryzować za pomocą menu rozwijanego narzędzia.

Wprowadź ręcznie powiązane dane narzędzia.

Albo te dane są już znane i udokumentowane z projektu, albo musisz sam je określić i wprowadzić!

Zapoznaj się z sekcją dotyczącą wprowadzania danych narzędzia w niniejszej instrukcji.

Po kliknięciu przycisku **[Upload Tool]** wprowadzone dane są aktualizowane do pamięci robota.

## 15.6.2 Obszar Wizard / Kreator

**TOOL WIZARD**

Only X,Y,Z  X,Y,Z,O,A,T Tool: 1

?

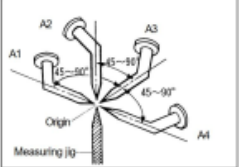
?

?

?

?

?

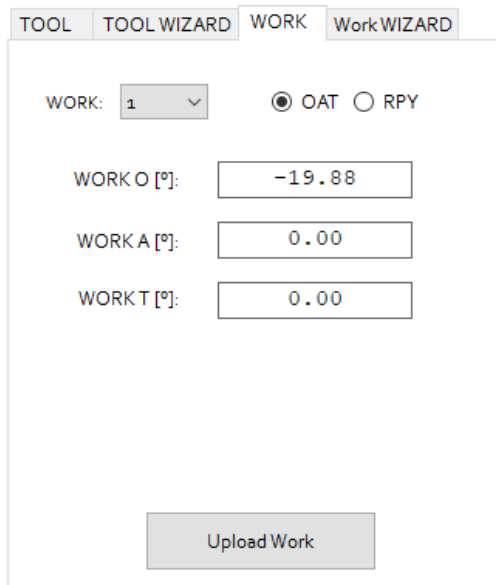


Ta sekcja umożliwia użytkownikowi obliczenie nowego TCP (TOOL CENTER POINT) metodą 4-punktową lub 6-punktową.

Metoda 4-punktowa pozwala na obliczenie wartości  $x, y, z$  danych narzędzia

Metoda 6-punktowa pozwala obliczyć wartości  $x, y, z$  i  $Rx, Ry, Rz$  (OAT) danych narzędzia. Zapoznaj się z sekcją dotyczącą obliczania narzędzia w niniejszej instrukcji.

### 15.6.3 WORK

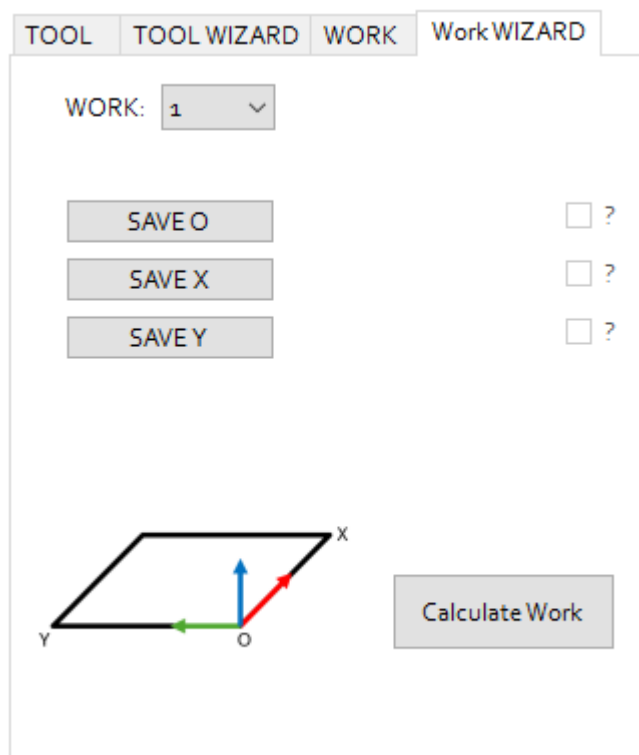


Ta sekcja umożliwia użytkownikowi sprawdzanie i zmianę wartości układu WORK.

Albo te dane są już znane i udokumentowane, albo muszą być określone i wprowadzone ręcznie.

Kliknięcie przycisku [Upload Work] powoduje zapisanie wprowadzonych danych do pamięci robota.

### 15.6.4 WORK WIZARD



Ta sekcja umożliwia użytkownikowi obliczenie nowego układu współrzędnych WORK przy użyciu metody 3-punktowej.

Naucz wszystkie 3 punkty na płaszczyźnie odniesienia, a następnie kliknij przycisk [Calculate Work]

## 15.6.5 Obszar Power OFF position / Pozycja wyłączenia

Power off position

JT1:	JT2:	JT3:	JT4:	JT5:	JT6:	JT7:	Save
0.000	-90.00	160.00	0.000	-90.00	0.000	0.000	Current
							Default

Ten obszar zawiera informacje o bezpiecznej pozycji wyłączenia.

- Przycisk [Save] służy do zapisywania ręcznie wprowadzonej pozycji.
- Użyj przycisku [Current], aby zapisać aktualną pozycję robota.
- [Default] służy do resetowania wartości do ustawień fabrycznych.

## 15.6.6 Obszar Zeroing order / Kolejność zerowania

Power off position Zeroing order

Default  Manual SAVE

JT1	JT2	JT3	JT4	JT5	JT6	JT7
2	3	4	5	6	7	1
<input checked="" type="checkbox"/> go to 0	<input checked="" type="checkbox"/> go to 0	<input checked="" type="checkbox"/> go to 0	<input checked="" type="checkbox"/> go to 0	<input checked="" type="checkbox"/> go to 0	<input checked="" type="checkbox"/> go to 0	<input checked="" type="checkbox"/> go to 0

Ten obszar zawiera informacje o kolejności zerowania osi. Ta sekcja pozwala użytkownikowi ustawić sekwencję zerowania dla wszystkich osi. Wybierz krok [1..7] dla wszystkich osi (wiele osi może być zerowanych w tym samym kroku) i wybierz, czy oś powinna lub nie powinna znajdować się w pozycji 0 (zero) po zakończeniu procesu.

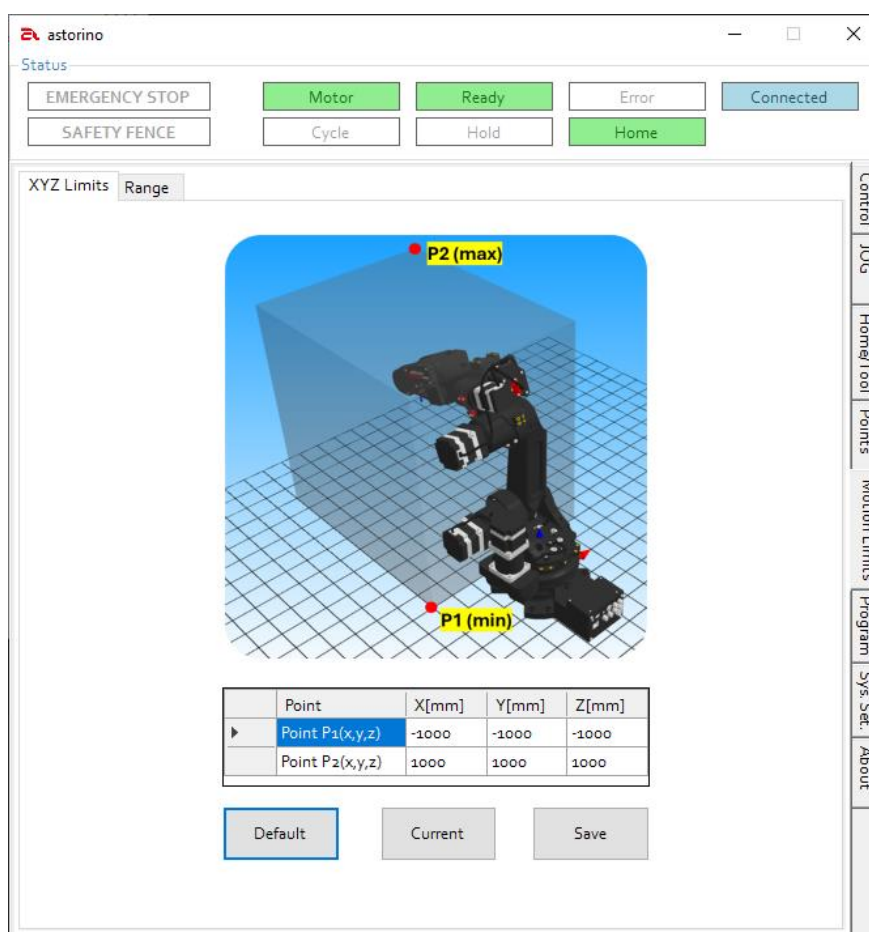
- Przycisk [Default] ustawia kolejność zerowania na domyślną kolejność,
- Przycisk [Manual] aktywuje sekcję ustawień ręcznych,
- Przycisk [Save] służy do zapisywania ręcznie wprowadzonej pozycji.

## 15.7 Karta Moving Area / Przestrzeń Robocza

W karcie [Moving Area] można zdefiniować dozwolony obszar roboczy robota Astorino.

Aby móc modyfikować dozwoloną przestrzeń pracy należy wejść na wyższy poziom dostępu. Aby to zrobić w Terminalu robota należy wpisać polecenie „z\_user 3”

### 15.7.1 XYZ Limits



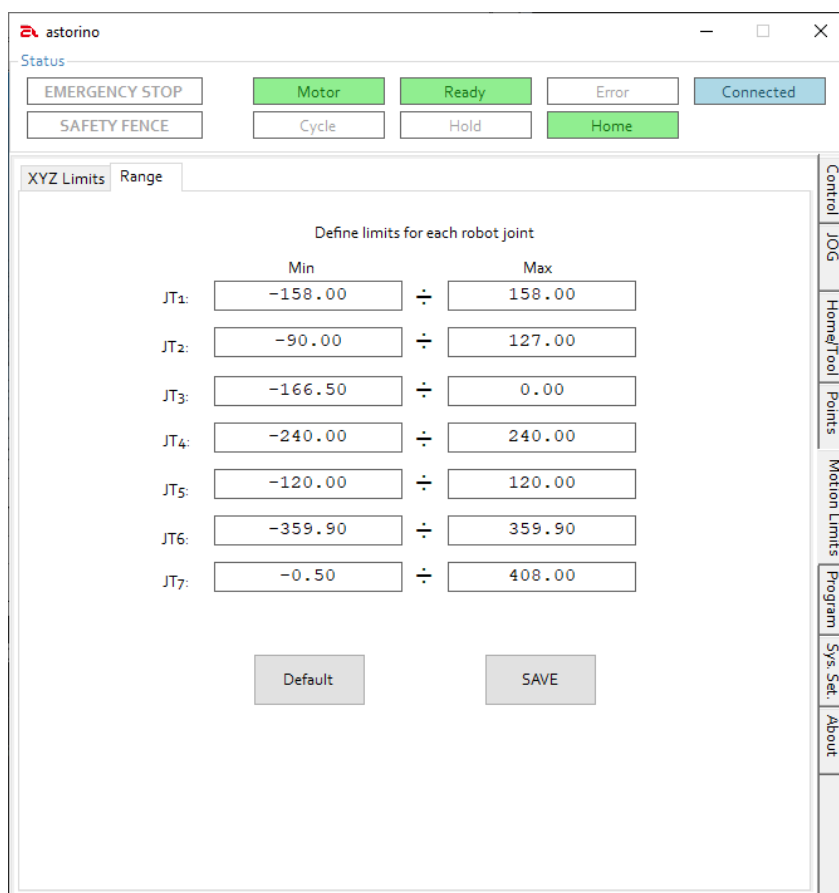
Za pomocą dwóch punktów **P1** i **P2** tworzona jest wirtualna prostokątna objętość, która określa obszar, w którym robot może się poruszać i którego nie może opuścić.

Punkt **P1** – pokazuje minimalne wartości,

Punkt **P2** – pokazuje maksymalne wartości

- za pomocą [Get Current] bieżąca pozycja robota jest rejestrowana i przechowywana w wybranym wierszu tabeli pozycji.
- Strefę można również zdefiniować ręcznie i wprowadzić wartości.
- za pomocą [SAVE] dane są przenoszone do pamięci robota.
- przycisk [Default] resetuje wartości do ustawień fabrycznych.

## 15.7.2 Range



The screenshot shows the 'astorino' software window. At the top, there is a 'Status' section with buttons for 'EMERGENCY STOP', 'SAFETY FENCE', 'Motor', 'Ready', 'Error', 'Connected', 'Cycle', 'Hold', and 'Home'. Below this is the 'XYZ Limits' section, with a sub-tab 'Range'. The main area is titled 'Define limits for each robot joint' and contains a table for configuring joint limits. The table has two columns: 'Min' and 'Max', with a division symbol between them. Below the table are 'Default' and 'SAVE' buttons. A vertical sidebar on the right contains buttons for 'Control', 'JOG', 'Home/Tool', 'Points', 'Motion Limits', 'Program', 'Sys. Set.', and 'About'.

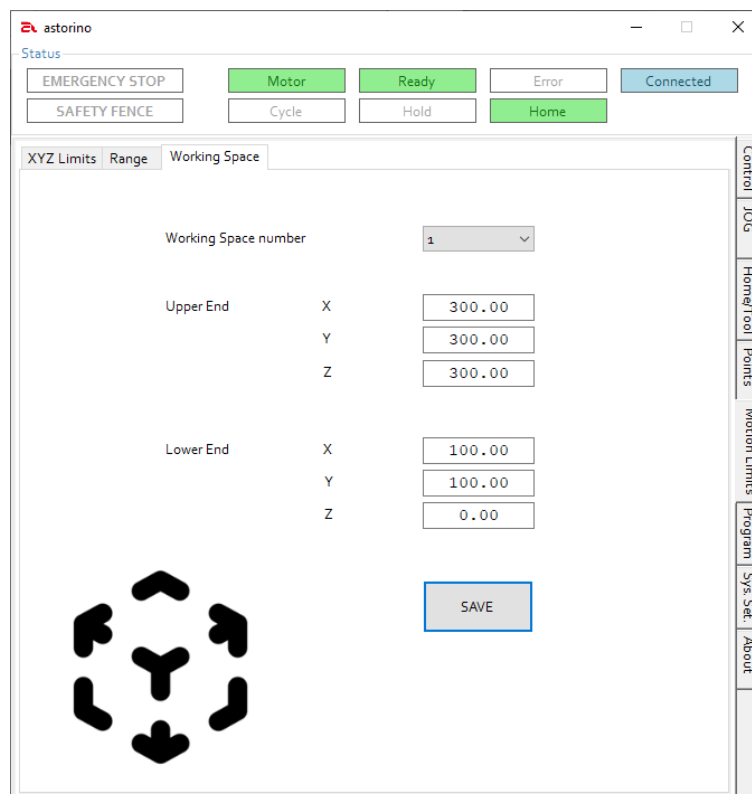
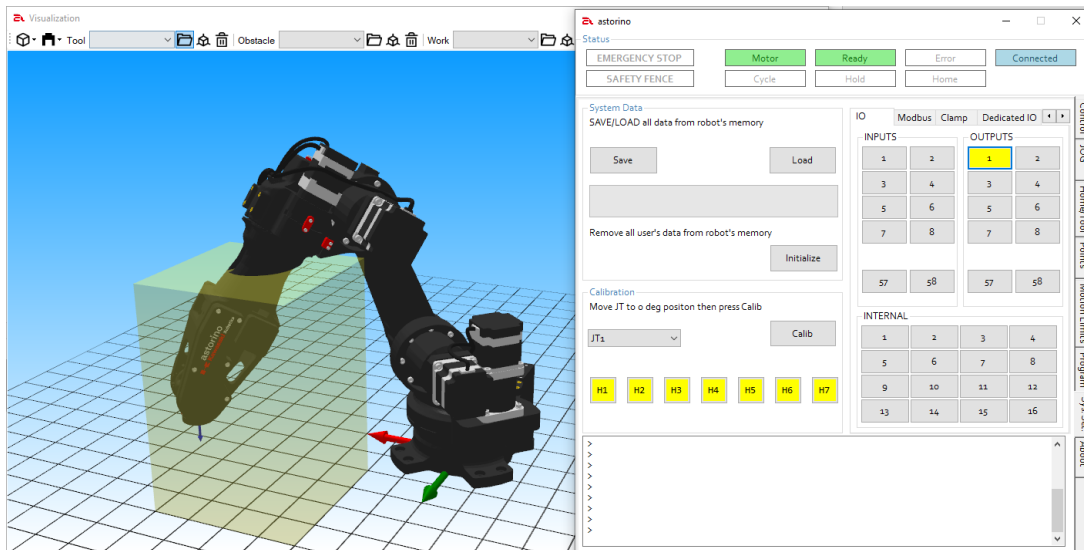
	Min		Max
JT1:	-158.00	÷	158.00
JT2:	-90.00	÷	127.00
JT3:	-166.50	÷	0.00
JT4:	-240.00	÷	240.00
JT5:	-120.00	÷	120.00
JT6:	-359.90	÷	359.90
JT7:	-0.50	÷	408.00

Na tej zakładce użytkownik może ustawić granice ruchu wszystkich osi osobno. Wpisz maksymalną i minimalną wartość kąta każdego złącza, a następnie naciśnij [SAVE], aby przesłać zmiany do robota. Naciśnięcie przycisku [Default] spowoduje przywrócenie ustawień fabrycznych zakresu osi.

## ASTORINO Instrukcja obsługi

### 15.7.3 Working Space

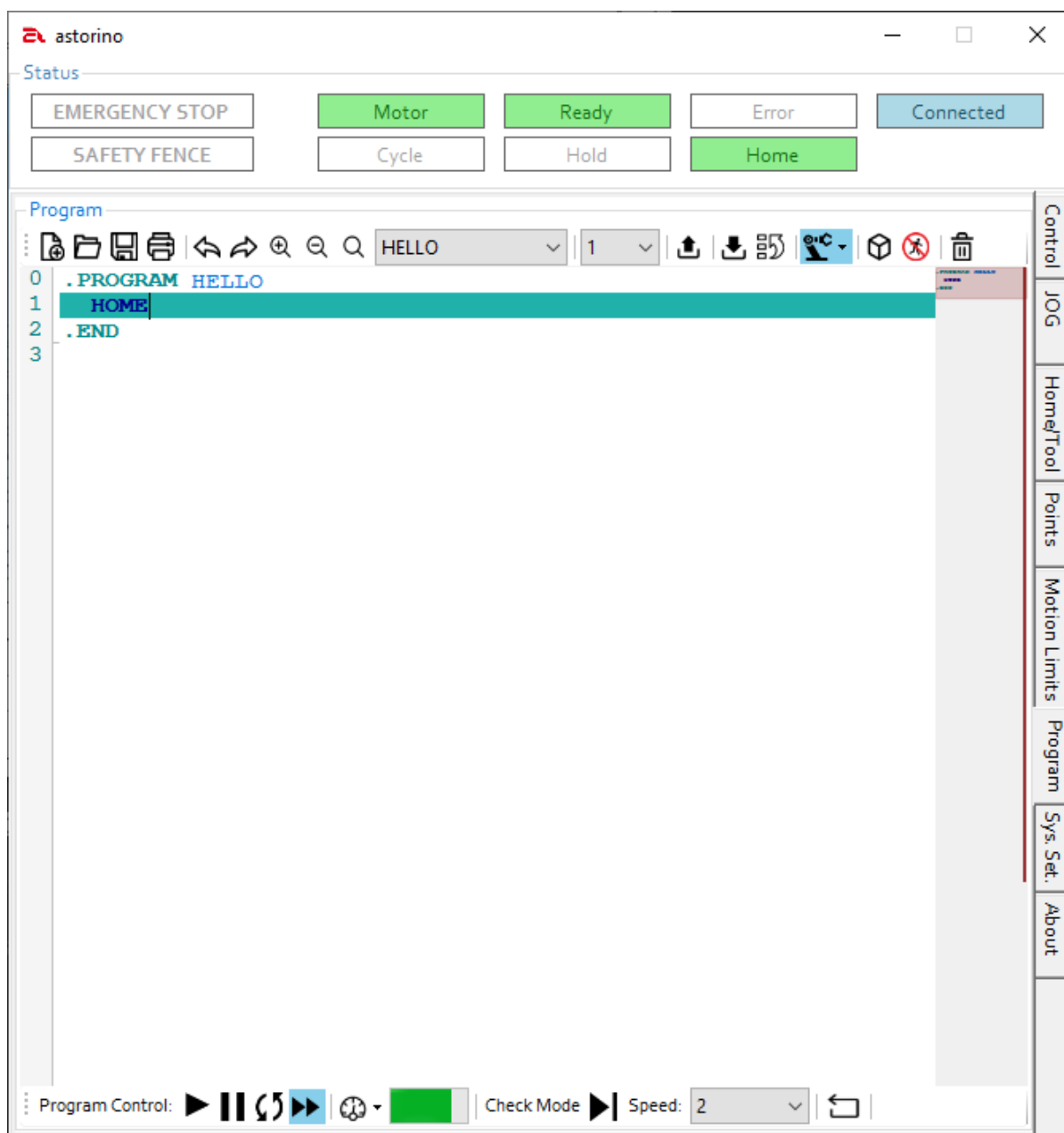
Przestrzeń robocza to wirtualny obszar, który może być wykorzystany do informowania innych urządzeń o położeniu ramienia robota. Przestrzeń robocza wykorzystuje do pracy dedykowane sygnały.



Na tej zakładce użytkownik może ustawić obszary Przestrzeni roboczej. Wpisz wartości górnego i dolnego końca każdej przestrzeni roboczej, a następnie naciśnij [ZAPISZ], aby przesłać zmiany do robota. Możesz ustawić do czterech różnych przestrzeni roboczych.



## 15.8 Karta Program

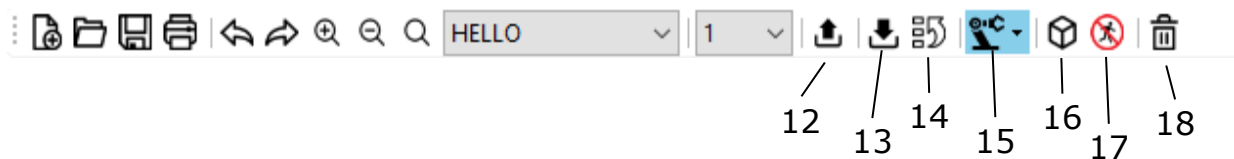


W tej zakładce możesz zaprogramować robota w uproszczonej wersji języka Kawasaki AS.

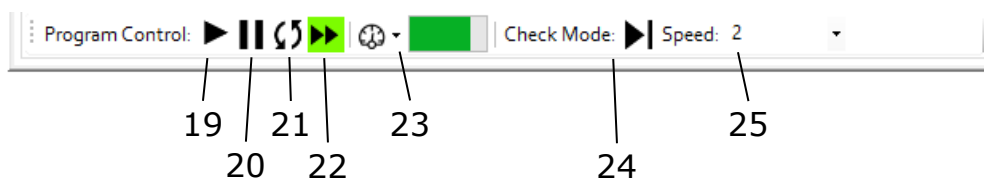
## ASTORINO Instrukcja obsługi



1. Tworzenie nowego programu
2. Wybierz program z komputera i załaduj go na robota
3. Zapisz wybrany program jako plik \*.pg na komputerze PC
4. Drukuj kod programu
5. Cofnij,
6. Ponów,
7. Powiększenie zawartości okna programu (zoom)
8. Zmniejszenie zawartości okna programu (zoom)
9. Resetowanie powiększenia do ustawień domyślnych
10. Lista wyboru programu,
11. Lista wyboru kroku wybranego programu,



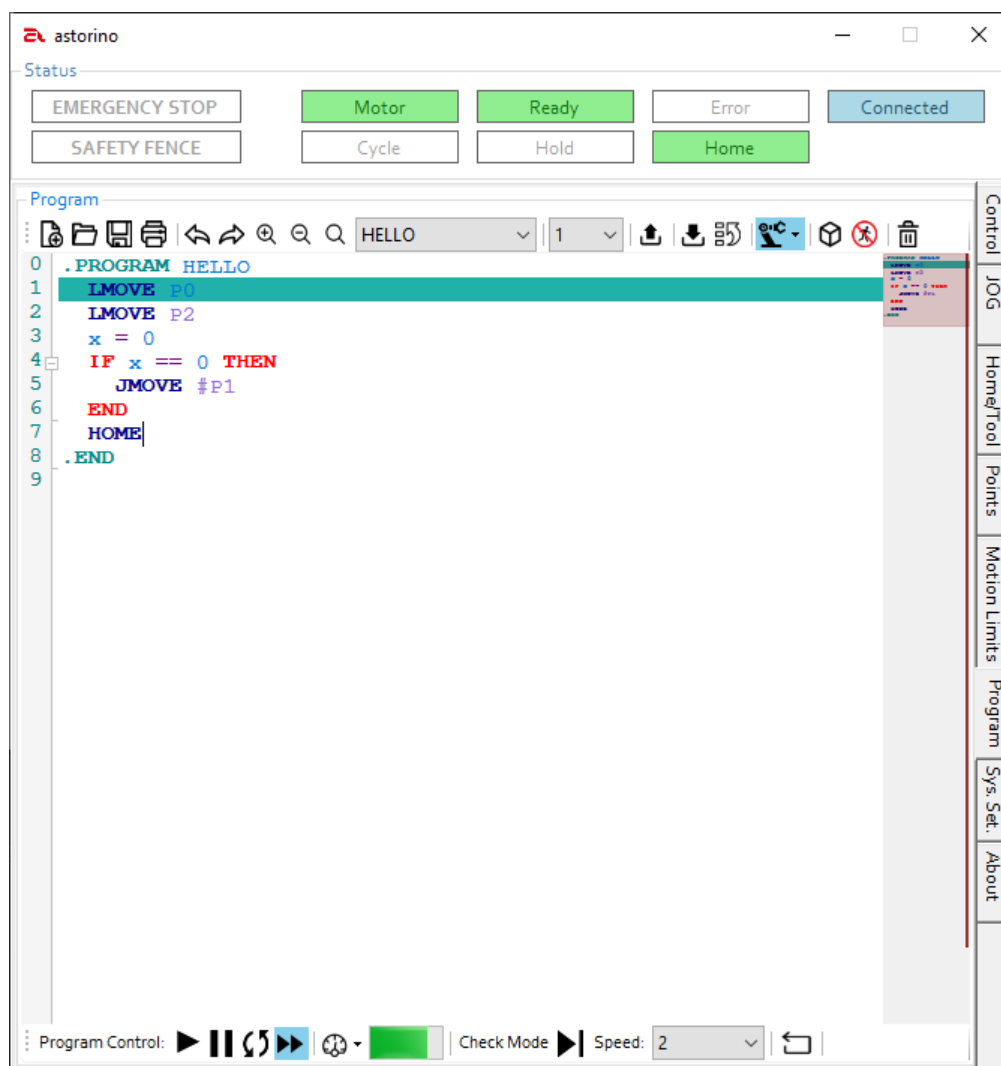
12. Wgrywa program do sterownika robota (PC ⇌ Astorino)
13. Pobiera wybrany program ze sterownika robota
14. Ustawia wybrany program jako program startowy \*
15. Przełączanie tryb pracy TEACH / REPEAT (ręczny / automatyczny)
16. Otwórz okno wizualizacji
17. Aktywuje tryb DryRun (bez ruchu robota)
18. Usuwa zaznaczony program



19. Uruchomienie lub zatrzymanie programu
20. HOLD - zatrzymuje uruchomiony program
21. Repeat Continues - aktywuje tryb pętli programu
22. Step Continues - aktywuje tryb pojedynczego kroku
23. Monitor speed - prędkość monitorowania
24. Check mode - wyświetla aktualnie aktywny tryb krokowy lub ciągły.  
Przycisk Next Step – pozwala na przejście do kolejnych kroków w trybie krokowym. Działa także, jako przycisk GO – wykonuje aktualnie zaznaczony krok programu,
25. Ustawianie i zmiana prędkości w trybie TEACH

## ASTORINO Instrukcja obsługi

\* Po włączeniu Astorino program zdefiniowany jako program startowy jest ładowany do pamięci roboczej sterownika robota i jest bezpośrednio gotowy do uruchomienia.



Przykładowy program o nazwie *hello*

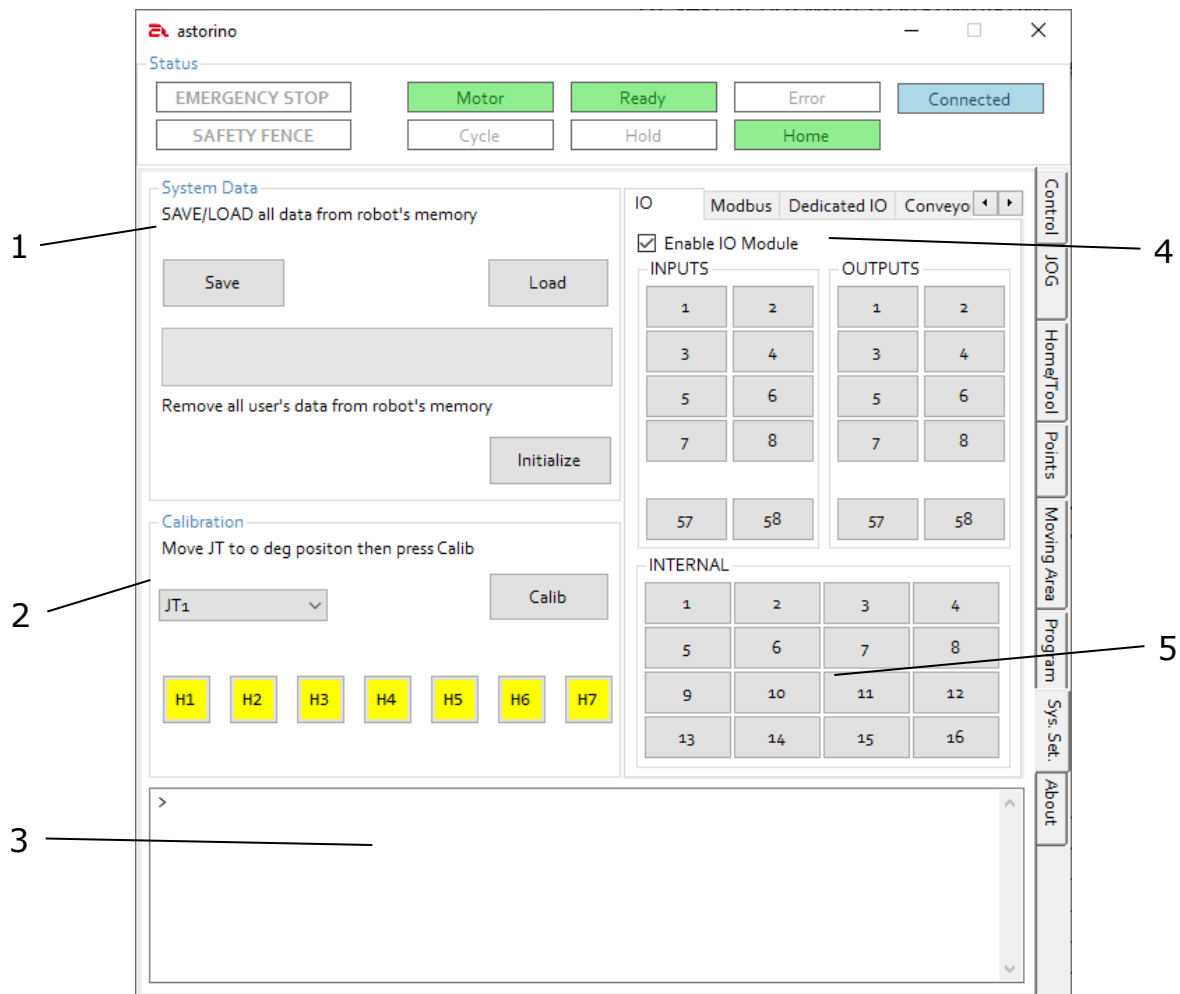
Jeśli program zostanie wykonany, robot porusza się do punktu **P1** w linii prostej (LinearMOVE).

Po osiągnięciu punktu ponownie porusza się po prostej ścieżce do **P2**. Zmiennej *x* przypisano teraz wartość 0.

Pętla IF sprawdza, czy *x* ma wartość 0.

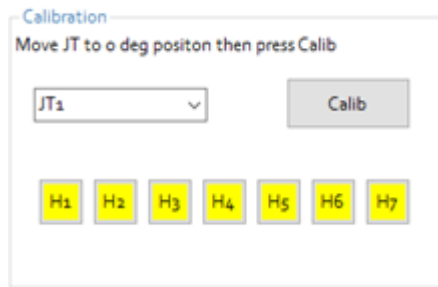
W takim przypadku wykonywane jest polecenie HOME, a robot przechodzi do pozycji domowej.

## 15.9 Karta System Settings / Ustawienia Systemu



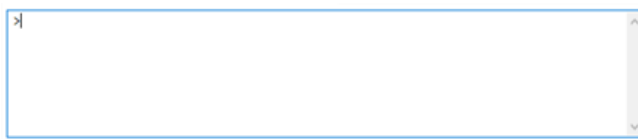
System Data	Zapisz/Załaduj wszystkie dane robota, lub zainicjalizuj pamięć robota.
Calibration	Kalibracja ramienia robota wymagana tylko wtedy, gdy dane na karcie SD zostaną usunięte, karta zostanie uszkodzona i wymieniona lub robot zostanie zdemontowany.
Terminal	Polecenia użytkownika i wyświetlanie danych.
IO, Dedicated IO, Conveyor...	Status wejść i wyjść (I/O), Skonfiguruj we/wy, inne ustawienia.
INTERNAL	Sterowanie sygnałami wewnętrznymi.

## 15.10 Obszar Calibration / Kalibracja



Ta sekcja pozwala użytkownikowi skalibrować robota, a także sprawdzić czujniki magnetyczne, które są zainstalowane na wszystkich osiach. Jeśli przyciski o nazwie Hx, gdzie x jest 1..7 świeci na żółto, to czujnik magnetyczny widzi magnes.

## 15.11 Pole Terminal



Terminal służy do wyświetlania informacji z robota, ale także do wydawania poleceń robotowi.

Wszystkie polecenia ruchu, takie jak LMOVE, HOME itp. Muszą być poprzedzone słowem "DO", a robot musi być GOTOWY i w trybie REPEAT. Na przykład "DO LMOVE P1"

Możesz także użyć terminala do odczytu wartości zmiennych (na przykład "PRINT x"), nauki punktów (na przykład HERE P1), ustawiania zmiennych (na przykład x = 10) i tak dalej.

Oto lista poleceń Terminala:

CPUTEMP	Pokazuje temperaturę procesora
FREE	Pokazuje dostępną pamięć RAM w %
ERESET	Resetuje błąd
ZPOWER ON	Włącza SILNIKI
ZPOWER OFF	Wyłącza SILNIKI
HOLD	Wstrzymuje aktualnie uruchomiony program
CONTINUE	Kontynuuje wstrzymany program
ZZERO x	Rozpoczyna zerowanie określonej osi - x
HALT	Wstrzymuje aktualnie uruchomiony program
EXECUTE x	Uruchamia aktualnie wybrany program, jeśli podana nazwa uruchamia wybrany program (x)
PRIME x	Wybiera wybrany program według nazwy
STOP	Zatrzymuje aktualnie uruchomiony program (Cykl)
REP_ONCE ON/OFF	Włączanie i wyłączanie funkcji REPEAT ONCE
STP_ONCE ON/OFF	Włączanie i wyłączanie STEP ONCE
STPNEXT	Wyzwala następny krok w trybie krokowym

## 15.11.1 Obszar Status i konfiguracja

### 15.11.1.1 IO

The screenshot shows the IO configuration window with the following sections:

- Enable IO Module:** A checked checkbox.
- INPUTS:** A grid of 16 buttons numbered 1 to 8, and 57 to 58.
- OUTPUTS:** A grid of 16 buttons numbered 1 to 8, and 57 to 58.
- INTERNAL:** A grid of 16 buttons numbered 1 to 16.

Red arrows point from the text on the right to the 'Enable IO Module' checkbox, the '58' output button, and the '10' internal button.

W tym oknie można zobaczyć stan WEJŚĆ sygnałów i włączyć/wyłączyć WYJŚCIA, klikając odpowiednie pola wyboru. □

Wybranie pola  **Enable IO Module** powoduje aktywację modułu We/Wy. Jeśli moduł IO jest uszkodzony, to pole jest automatycznie odznaczone.

WEJŚCIA 57,58 i WYJŚCIA 57,58 są zarezerwowane dla wersji B robota i znajdują się na ramieniu JT3

#### [ATTENTION]

Jeśli aktywujesz tę funkcję bez modułu podłączenia I/O do płyty głównej robota, wystąpi błąd!

**INTERNAL** – tutaj sprawdzasz lub wymuszasz status sygnałów wewnętrznych. ( **ON** , OFF)

## ASTORINO Instrukcja obsługi

### 15.11.1.2 MODBUS

IO Modbus Dedicated IO Conveyo

Fieldbus Inputs

9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56


Fieldbus Outputs

9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56

W tym oknie można zobaczyć stan wejść FIELDBUS i włączyć/wyłączyć wyjścia FIELDBUS klikając odpowiednie przyciski sprawdzania. Jeśli wejście lub wyjście jest włączone, przycisk świeci na żółto.

### 15.11.1.3 Clamp

IO Modbus Clamp Dedicated IO

Handling Clamp Signal Definition 

OUTPUT Signal for OFF

OUTPUT Signal for ON

W tym oknie zmienić można ustawienia Handling Clamp

Ustawienia clamp mają wpływ na polecenia OPENI i CLOSEI.

## ASTORINO Instrukcja obsługi

### 15.11.1.4 Sygnały dedykowane

Widok i konfiguracja dedykowanych sygnałów robota. Sygnały mają ustaloną, określoną funkcję lub instrukcję.

Dedykowane wejścia to specjalne sygnały wejściowe, takie jak "Reset" lub "Cycle Stop".

Dedykowane wyjścia to specjalne sygnały wyjściowe, takie jak "Motor ON" i "Error".

MZH (MOTOR ON -> ZEROING -> HOME) – jest to specjalna sekwencja przeznaczona do homowania robota z jednym bitem ze stanu początkowego po włączeniu zasilania.

### 15.11.1.5 Detekcja kolizji

B – wersja robota wyposażona jest w akcelerometr do wykrywania kolizji.

Tutaj użytkownik może zmienić progi wykrywania kolizji.

Pole wyboru [4G Mode] umożliwia włączenie/wyłączenie progów wysokiego poziomu

Przycisk [Save] zapisuje progi do pamięci robota

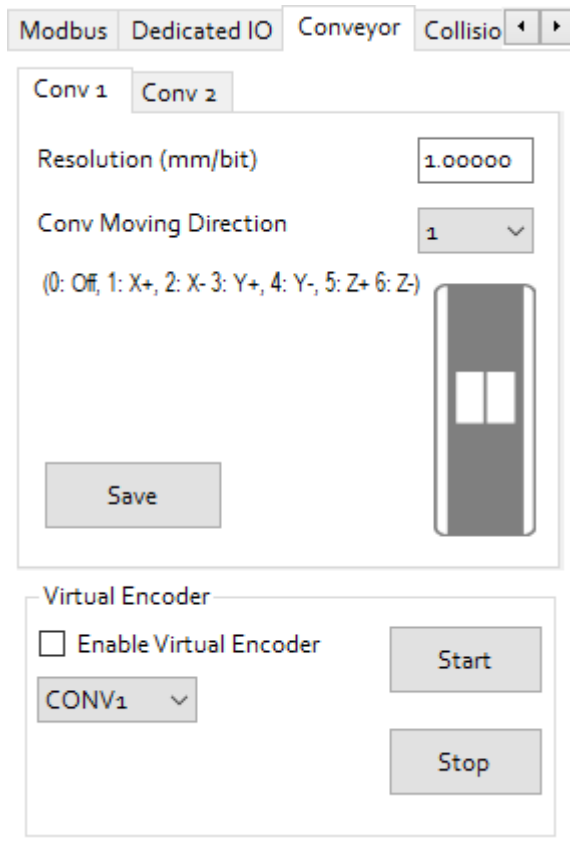
Przycisk [Calib ON/OFF] rozpoczyna lub zatrzymuje automatyczną kalibrację progów.

\*aby zmienić ustawienia wpisz w terminalu polecenie „Z\_USER 3”



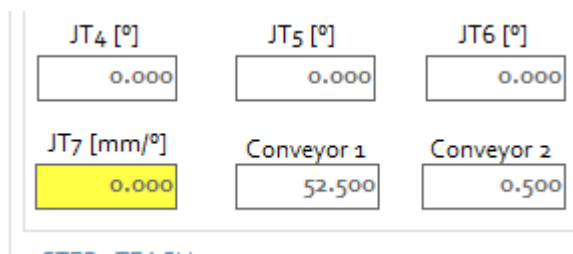
## ASTORINO Instrukcja obsługi

### 15.11.1.6 Taśmociąg



W tym miejscu można ustawić kierunek ruchu taśmociągu. Ustawiamy kierunek zgodnie z jedną osi układu BASE robota, a także rozdzielczość mm/bit

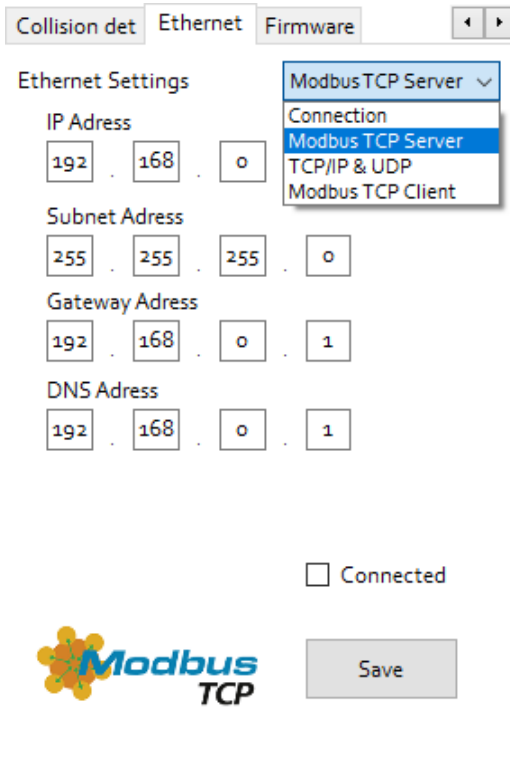
Można także załączyć wirtualny enkoder. Pozwala to na symulację aplikacji, które wykorzystują śledzenie tasmy. Wybieramy z listy, który taśmociąg chcemy uruchomić (CONV1 lub CONV2), załączmy checkbox (Enable Virtual Encoder), a następnie przyciskami [Start] lub [Stop] włączamy lub wyłączamy wirtualny enkoder.



Wartości wirtualnego jak i fizycznego enkodera wyświetlają się z zakładce JOG

## ASTORINO Instrukcja obsługi

### 15.11.1.7 Ethernet



Collision det Ethernet Firmware

Ethernet Settings


IP Address  
192 . 168 . 0

Subnet Address  
255 . 255 . 255 . 0

Gateway Address  
192 . 168 . 0 . 1

DNS Address  
192 . 168 . 0 . 1

Connected

 Save

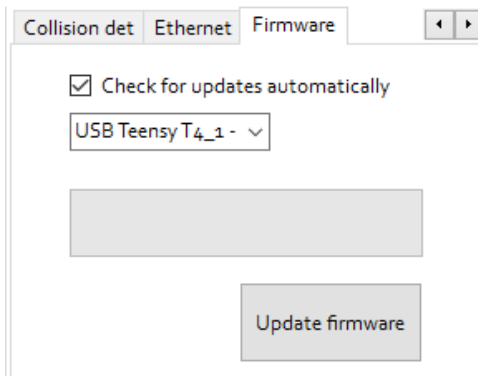
W tym obszarze można zmienić ustawienia komunikacji Ethernet.

Działanie portu Ethernet można ustawić na:

- Połączenie z oprogramowaniem Astorino
- Modbus TCP Serwer
- TCP/IP lub UDP
- Modbus TCP Klient

Przycisk [Save] zapisuje zmiany w pamięci kontrolera. Po zapisaniu wymagane jest ponowne uruchomienie robota.

### 15.11.1.8 Firmware



Collision det Ethernet Firmware

Check for updates automatically

USB Teensy T4\_1

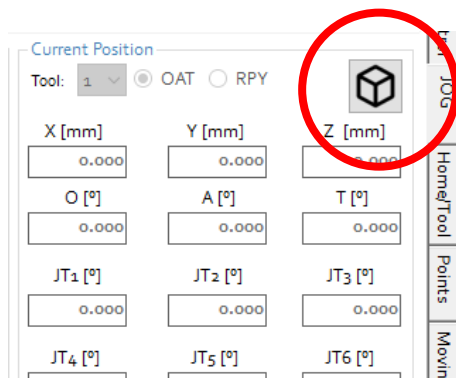
Update firmware

W tej podzakładce możesz zaktualizować oprogramowanie Astorino. A także ustawić czy oprogramowanie ma sprawdzać czy pojawiły się aktualizacje na serwerze.

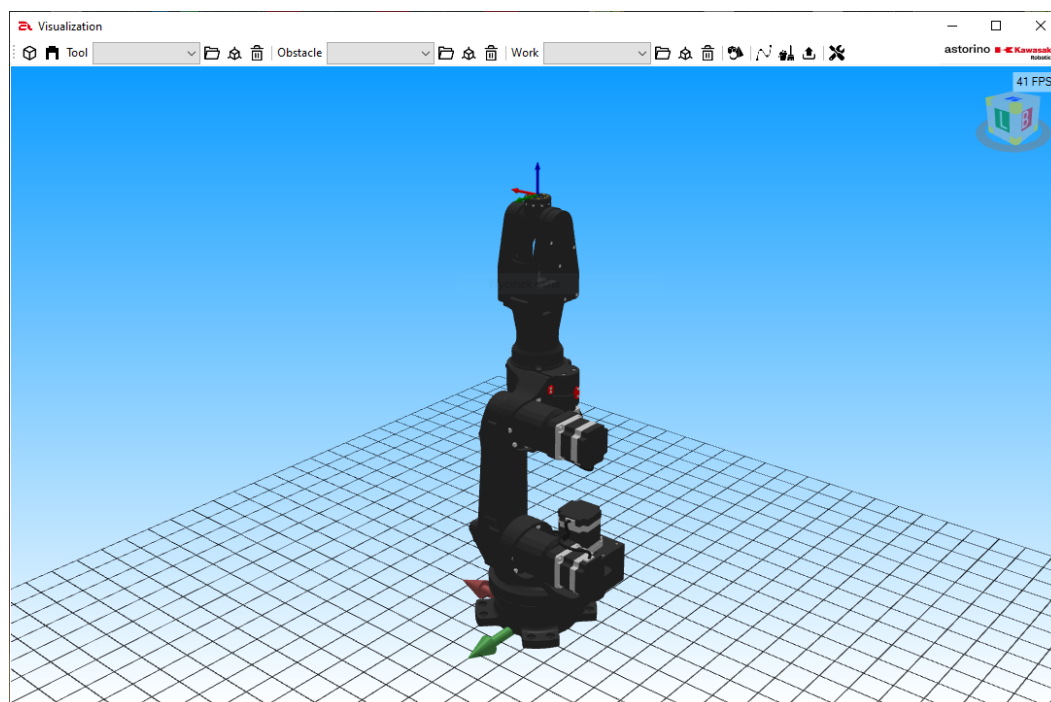
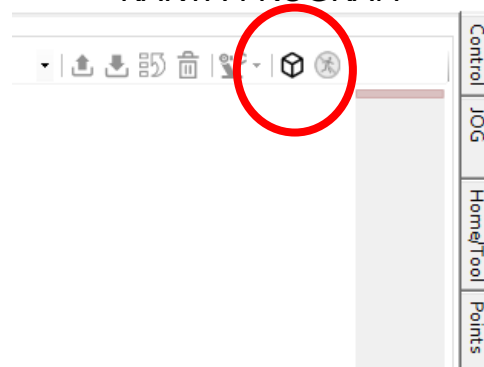
## 15.12 Okno Wizualizacji

Aby otworzyć okno wizualizacji i zobaczyć w czasie rzeczywistym działanie robota Astorino, należy kliknąć jeden z tych dwóch przycisków

KARTA JOG



KARTA PROGRAM



### 15.12.1 Obsługa okna wizualizacji

Okno wizualizacji pozwala na dodawanie obiektów 3D do sceny z robotem. Program obsługuje pliki typu stl i pozwala na dodawanie podstawowych kształtów trójwymiarowych. Każdy z obiektów można dodać jako jeden z trzech typów obiektów:

- Obstacle – obiekty tego typu są statycznymi obiektami sceny
- Work – obiekty tego typu mogą być przenoszone przez robota
- Tool – obiekty tego typu zawsze poruszają się zgodnie z kłasią robota.

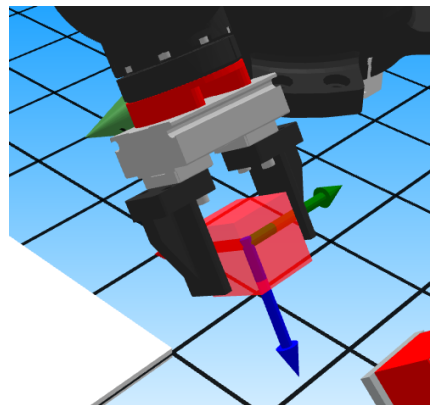
## ASTORINO Instrukcja obsługi

Menu okna wizualizacja składa się z następujących elementów:

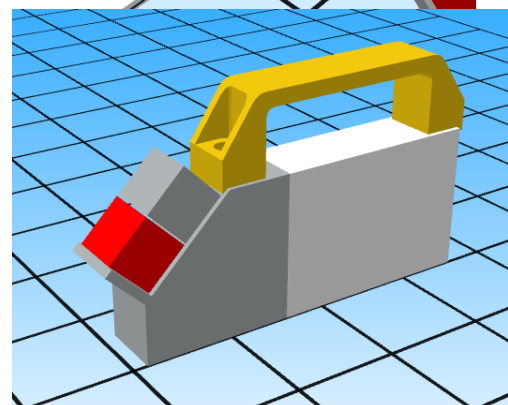
	Włącza lub wyłącza widok przestrzeni pracy robota (Working Space)
	Włącza lub wyłącza model 3D standardowego chwytaka
	Zawiera listę obiektów klasy Tool, Obstacle lub Work
	Otwiera plik stl i wczytuje go jako jeden z obiektów klasy Tool, Obstacle lub work
	Włącza menu modyfikacji obiektów, pozwala na zmianę pozycji obiektu lub zmianę jego koloru
	Usuwa zaznaczony aktualnie obiekt na liście rozwijanej
	Włącza menu generatora prostych kształtów 3D
	Włącza generowanie wizualizacji trajektorii robota
	Wyłącza i czyści wizualizację trajektorii robota
	Zapisuje do pliku .traj punkty wizualizacji trajektorii robota
	Włącza menu ustawień okna wizualizacji

### 15.12.2 Klasy obiektów

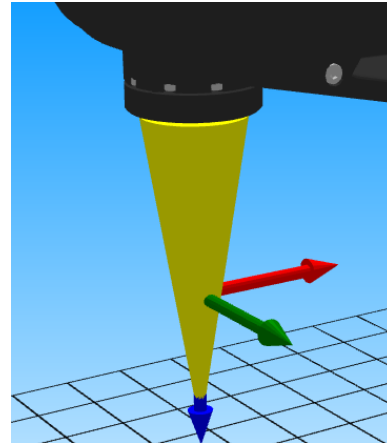
Work – obiekty klasy Work mogą być przenoszone przez robota. Aby obiekt został pochwycony punkt TCP musi się znajdować wewnątrz pobieranego obiektu, a sygnał sterujący musi być w stanie wysokim.



Obstacle – obiekty klasy Obstacle są statycznymi elementami wizualizacji. Pozwalają zbudować scenę wizualizacji, stanowią aspekt wizualny oraz potencjalne przeszkody.



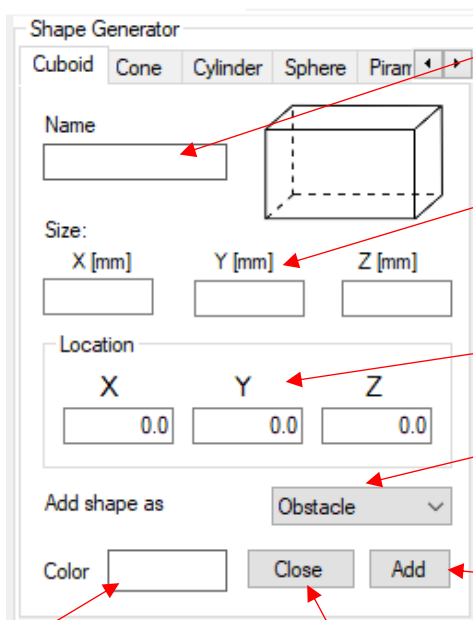
Tool – obiekty klasy Tool są to obiekty, które na stałe są przymocowane do końcówki robota. Dzięki tym obiektom można stworzyć swoje własne narzędzia, które są zamontowane na flaszach robota.



### 15.12.3 Generator prostych kształtów

Generator prostych kształtów trójwymiarowych pozwala na generowanie następujących elementów:

- Sześcian (kostka),
- Prostopadłościan,
- Stożek,
- Cylinder,
- Sfera,
- Piramida,
- Rura



Nazwa obiektu, gdy nie wpisana – przypisuje obiektowi kolejną wolną nazwę automatycznie.

Rozmiar obiektu, w zależności od figury należy podać od 1 do 3 parametrów.

Pozycja pod jaką ma zostać stworzona figura.

Wybór klasy obiektu jaki ma zostać stworzony: Tool, Obstacle, Work.

Dodanie obiektu do wizualizacji

Wybór koloru obiektu

Zamknięcie menu generatora

## ASTORINO Instrukcja obsługi

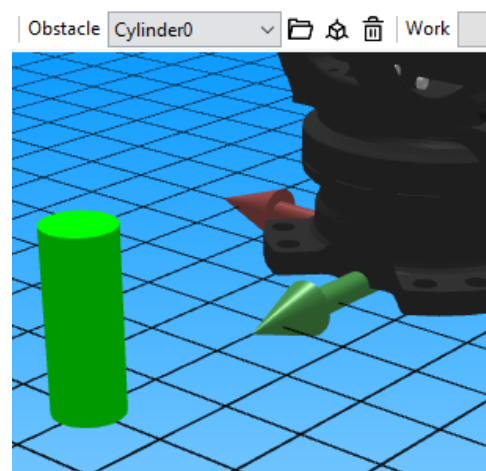
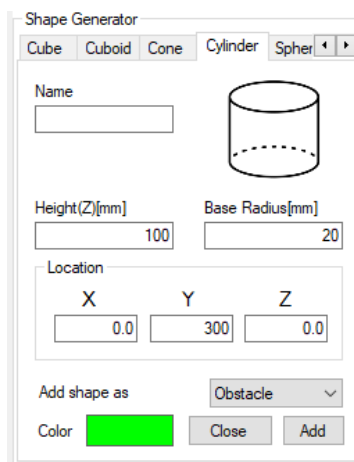
---

### Przykład

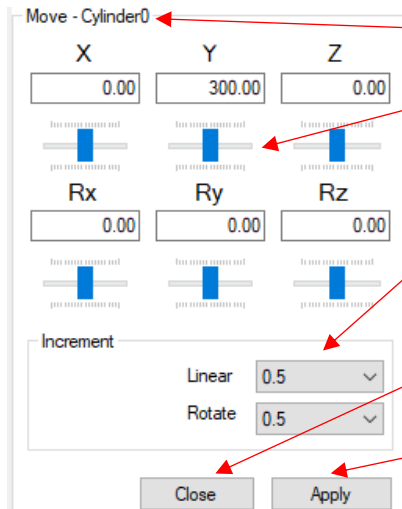
Dodać do wizualizacji cylinder klasy obstacle o parametrach:

- 100 mm wysokości,
- Promień podstawy 20mm,
- Kolor zielony
- Pozycja początkowa (0,300,0 [x,y,z])
- Nazwa dowolna

Aby dodać taki obiekt należy wpisać następujące dane w menu generatora i potwierdzić przyciskiem [ADD]. Obiekt zostanie dodany.



## 15.12.4 Menu modyfikacji obiektów



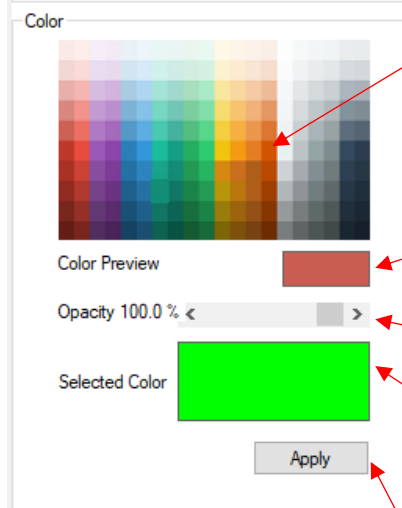
Nazwa aktualnie modyfikowanej figury.

Suwaki oraz pola tekstowe pozwalające zmienić pozycję obiektu na wizualizacji.

Ustawienie rozdzielczości powyższych suwaków

Zamknięcie okna modyfikacji.

Zatwierdzenie zmian pozycji obiektu.



Obszar szybkiego wyboru koloru, kliknięcie przyciskiem myszy na konkretnym kolorze pozwoli go wybrać jako kolor modyfikowanego obiektu

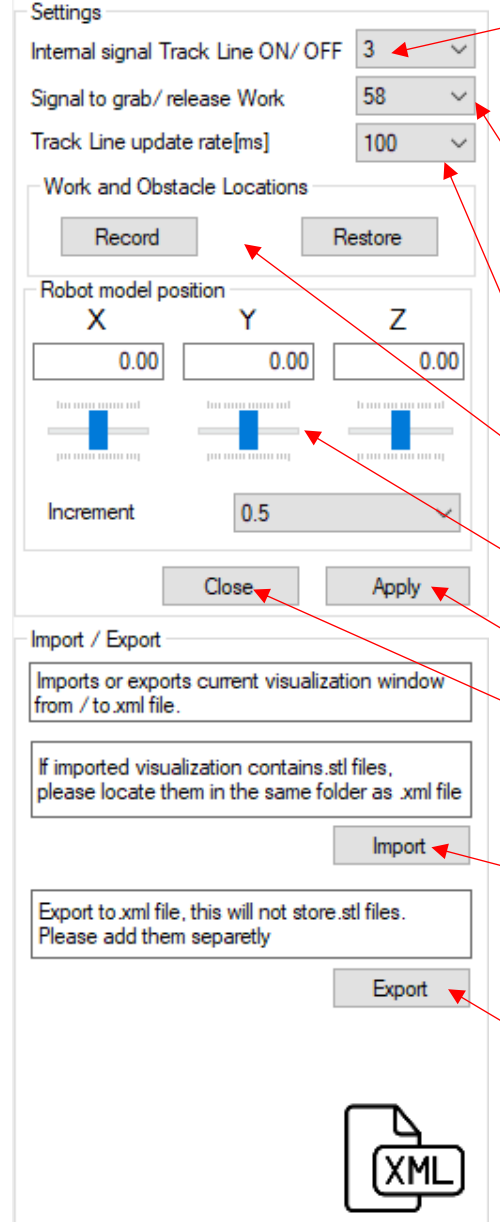
Okno podglądu wybrane koloru.

Ustawienie nieprzeźroczystości obiektu.

Aktualnie wybrane kolor, kliknięcie w ten obszar otworzy okno zaawansowanego wyboru koloru

Zatwierdzenie zmian.

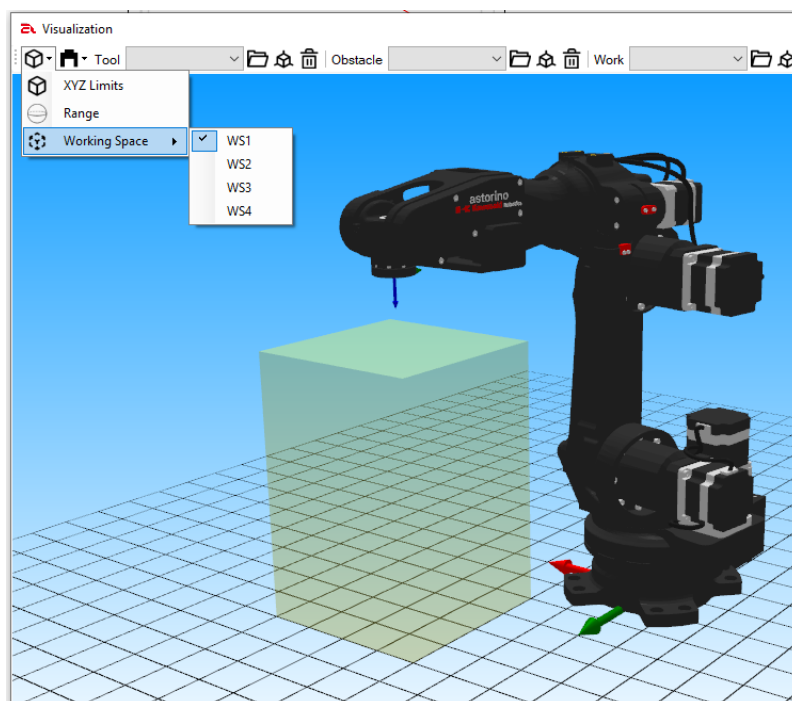
## 15.12.5 Menu ustawień wizualizacji

	<p>Sygnal wewnętrzny (z puli 2001-2016), który włącza lub wyłącza generowanie punktów trajektorii na wizualizacji.</p> <p>Sygnal wyjściowy sterujący łapaniem obiektów klasy Work w wizualizacji przez robota.</p> <p>Czas co jaki tworzony jest kolejny punkt wizualizacji trajektorii.</p> <p>Zapisanie oraz przywrócenie pozycji elementów znajdujących się na wizualizacji.</p> <p>Zmiana pozycji robota na wizualizacji.</p> <p>Zatwierdzenie zmian.</p> <p>Zamknięcie menu ustawień.</p> <p>Import wizualizacji z pliku xml.</p> <p>Eksport wizualizacji do pliki xml. Eksport nie zapisuje otworzonych plików stl. Tylko ich nazwy. Pliki należy skopiować osobno.</p>
--	---



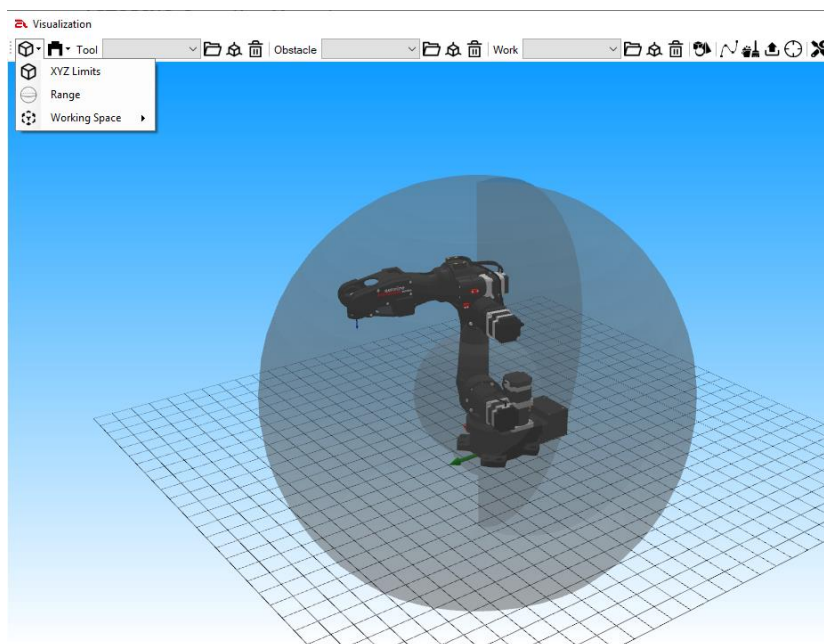
### 15.12.6 Wizualizacja Working Space

Okno wizualizacji pozwala na wyświetlenie wszystkich czterech Przestrzeni Roboczych. Wybierz, która z nich ma zostać dodana do sceny, a następnie kliknij [WS1.. 4]. Ponowne kliknięcie spowoduje ukrycie modelu 3D przestrzeni roboczej.



### 15.12.7 Wizualizacja zasięgu pracy robota

Okno wizualizacji pozwala na wyświetlenie zasięgu roboczego robota. Kliknij przycisk [Range], aby dodać lub usunąć model 3D zakresu roboczego.



## 15.13 Karta About / Informacje

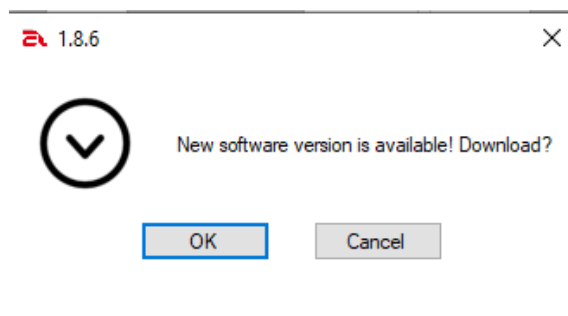


Ta zakładka pokazuje aktualną wersję oprogramowania Astorino i kompatybilnego oprogramowania sprzętowego z aktualnym oprogramowaniem Astorino.

## 15.14 Aktualizacja oprogramowania sprzętowego

### 15.14.1 Podstawowe informacje

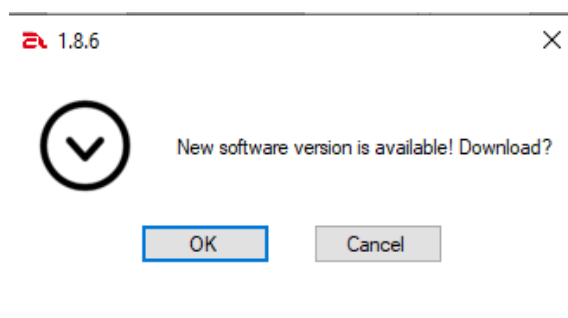
Oprogramowanie Astorino po uruchomieniu automatycznie sprawdzi, czy dostępna jest nowa wersja, a jeśli tak, poinformuje o tym użytkownika.



Kliknięcie przycisku [OK] spowoduje pobranie nowej wersji do określonej przez użytkownika lokalizacji na dysku twardym.

Następnie użytkownik musi odinstalować starą wersję z komputera i zainstalować nową.

Po połączeniu z robotem oprogramowanie astorino sprawdzi, czy oprogramowanie robota jest aktualne. Jeśli nowe oprogramowanie układowe jest dostępne, aplikacja poinformuje o tym użytkownika.



Kliknięcie przycisku [OK] spowoduje pobranie do nowej lokalizacji na dysku twardym określonej przez użytkownika.

Możesz pobrać najnowszą wersję oprogramowania z serwera FTP Kawasaki Robotics: <https://ftp.kawasakirobot.de/Software/Astorino/>

lub skontaktuj się z pomocą techniczną:

[Astorino@astor.com.pl](mailto:Astorino@astor.com.pl) / [Tech-Support@kawasakirobot.de](mailto:Tech-Support@kawasakirobot.de)

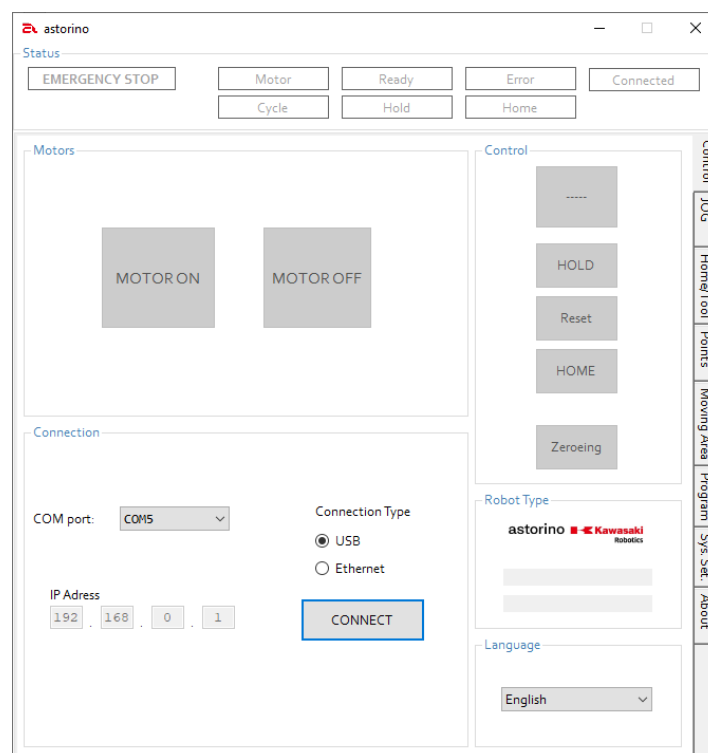
## 15.14.2 Procedura aktualizacji

Aby zaktualizować oprogramowanie sprzętowe, uruchom oprogramowanie Astorino.

### [UWAGA]

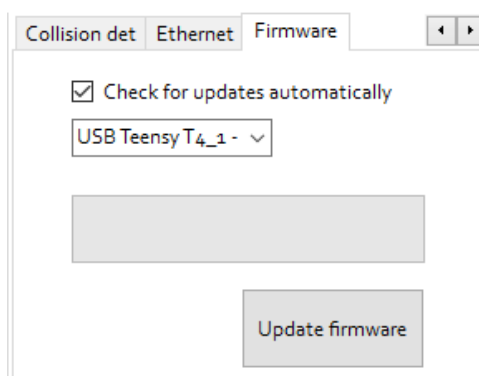
Upewnij się, że oprogramowanie **nie** jest podłączone do robota. Silniki muszą być wyłączone!

Przerwanie procesu może uszkodzić procesor, nie wyłączaj robota podczas procedury aktualizacji!



Przejdź do [System Setting], a tam do obszaru konfiguracji IO.

Klikaj symbol strzałki w prawo, aż pojawi się podzakładka [Firmware]

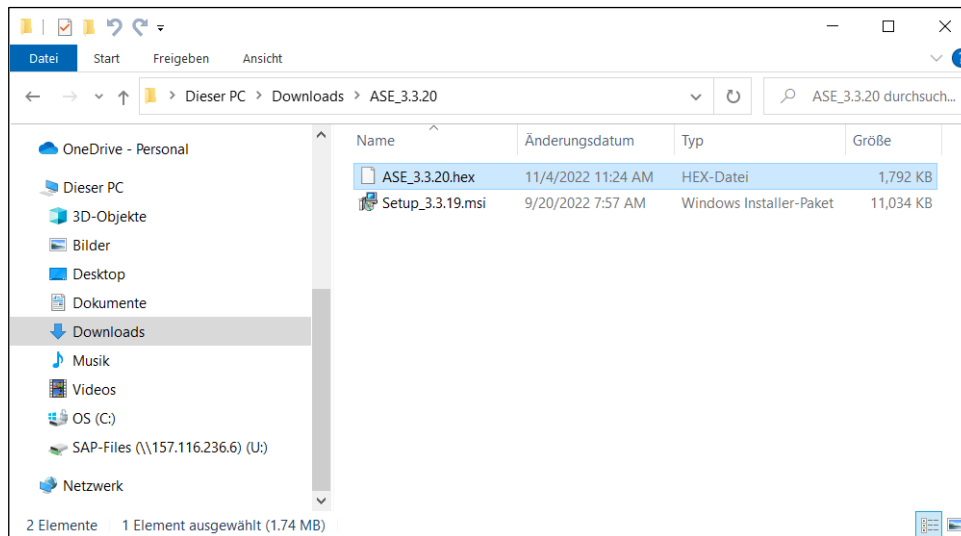


Kliknij przycisk [Update firmware], aby otworzyć okno wyboru plików.

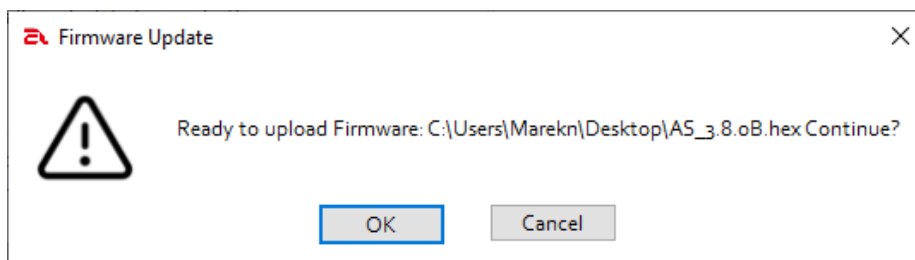
Wybierz plik \*.hex, który zawiera nowe oprogramowanie.

## ASTORINO Instrukcja obsługi

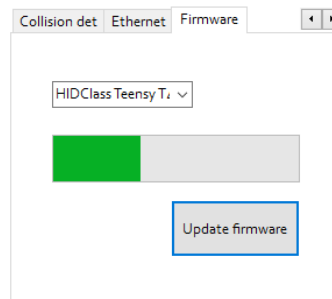
Okno wyboru pliku:



Potwierdź, że chcesz przesłać nowe oprogramowanie do pamięci robota:

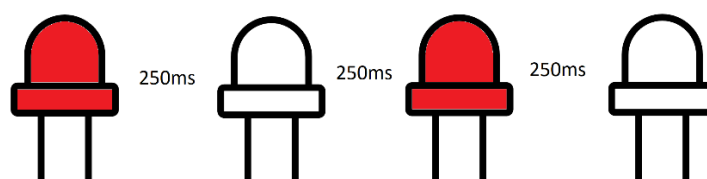


Zostanie przeprowadzona aktualizacja oprogramowania sprzętowego.



Po zainstalowaniu oprogramowania układowego obserwuj czerwoną diodę LED (Error) na podstawie robota.

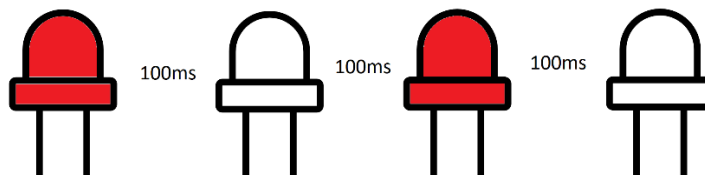
Jeśli dioda zacznie migać powoli (około 2 razy na sekundę).



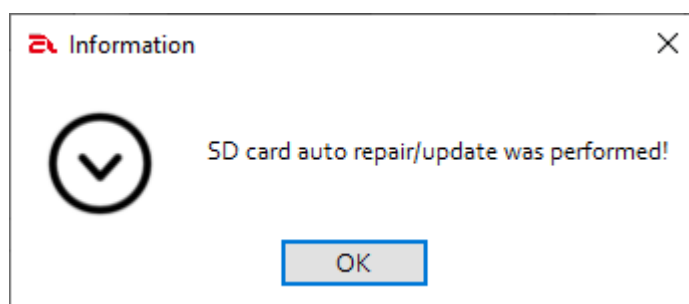
## ASTORINO Instrukcja obsługi

Wyłącz robota i włącz go ponownie. Jest to błąd karty SD wewnątrz podstawy robota, procesor nie mógł ponownie uruchomić karty po aktualizacji oprogramowania układowego. Zresetowanie zasilania rozwiązuje problem.

Jeśli dioda LED zacznie migać szybko (około 5 razy na sekundę).

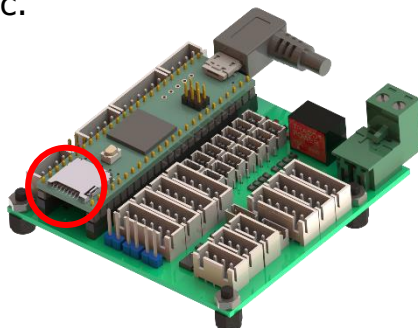


Oznacza to, że robot musi zaktualizować dane na karcie SD. Procedura wykonana się samoczynnie. Po zakończonej procedurze czerwoną diodą LED się wyłączy. A po podłączeniu do komputera pojawi się komunikat o zakończeniu procedury naprawy/aktualizacji danych na karcie SD



### 15.14.3 Odzyskiwanie po awarii aktualizacji

Jeśli proces został przerwany, procesor może nie działać, a system Windows nie wykryje robota. Możesz zresetować procesor do ustawień fabrycznych, naciskając biały przycisk reset na procesorze przez **13s do 17s**, a następnie czerwona dioda LED na jednostce procesora zacznie migać. Po zakończeniu procedury twardego pomarańczowa dioda LED na procesorze będzie migać powoli, a czerwona dioda LED (Error) na podstawie robota również zacznie powoli migać.



Biały przycisk znajduje się na płycie procesora wewnątrz podstawy robota. Aby uzyskać do niego dostęp, należy odkręcić i zdjąć tylną pokrywę górną.

## ASTORINO Instrukcja obsługi

---



Bądź bardzo ostrożny/a, robiąc to! Nie używaj żadnych metalowych przedmiotów wewnątrz podstawy robota podczas przywracania ustawień fabrycznych procesora!

## 15.15 Język AS

Astorino można zaprogramować przy użyciu podstawowej wersji języka Kawasaki AS-language, który jest używany we wszystkich robotach przemysłowych Kawasaki Robotics. Aby dowiedzieć się więcej o języku proszę zapoznać się z dokumencją „Instrukcja języka AS”

Poniższa instrukcja w skrócie opisuje podstawowe funkcje używane do programowania pracy robota.

### Aktualna lista obsługiwanych poleceń i funkcji:

(x,y,z reprezentują wartości - np. SPEED 100 ALWAYS<sub>p</sub> dla punktów lub nazw punktów - np. JMOVE P10)

Nazwa	Opis
ACCEL x	przyspieszenie robota w % dla polecenia następnego ruchu
ACCEL x ALWAYS	przyspieszenie robota w %
ALIGN	Wyrównywanie osi Z narzędzia do najbliższej osi BASE
C1MOVE p	określa punkt pośredni interpolacji kołowej
C2MOVE p	przesuwa robota do punktu p w interpolacji kołowej, przechodząc przez punkt określony w poleceniu C1MOVE p; przed poleceniem C2MOVE należy użyć polecenia C1MOVE
CVCOOPJT x	Ustawia współpracę między przenośnikiem numer 1 lub 2
CVDELAY x	Robot utrzymuje aktualną pozycję przenośnika przez czas x
CVLAPPRO p,x	Porusza się w kierunku Z narzędzia na danej odległości x od punktu p liniowo ze śledzeniem przenośnika
CVLDEPART x	przesuwa robota z bieżącej pozycji w określonej odległości x od bieżącej pozycji wzdłuż osi Z narzędzia ze śledzeniem przenośnika
CVLMOVE p	Ruch liniowy do punktu p ze śledzeniem przenośnika
CVRESET x	Resetuje dane przenośnika do wartości x



## ASTORINO Instrukcja obsługi

CVWAIT x	Czeka, aż przENOŚNIK dotrze do wartości x
DECEL x	Hamowanie w % dla polecenia następnego ruchu
DECEL x ALWAYS	Hamowanie w %
DISTANCE(p,p)	Oblicza odległość między dwoma punktami
DLYSIG x,y	Aktywuje sygnał X (1-58 lub Int. 2001-2016) po upływie czasu Y w sekundach
DRIVE x,y,z	Porusza pojedynczą osią o x- oś, y- kąt, z -prędkość
DRAW x,y,z	ruch liniowy względem x,y,z zgodnie z BASE
\$DECODE(x,y)	Funkcja wyszukuje separator y w ciągu x i wyodrębnia wszystkie znaki znajdujące się przed separatorem. Znaki te są ponownie wyprowadzane jako ciąg znaków i jednocześnie usuwane z oryginalnego ciągu!
\$ENCODE(x)	Zmienia liczbę na ciąg znaków
ERESET	Reset błędu
EXISTCOM	Stan gotowości danych komunikacyjnych HOST (Serial)
HERE p	Zapisz aktualną pozycję robota do punktu P
HOME	przesuwa robota do pozycji HOME
INRANGE(p)	Sprawdza, czy punkt znajduje się w zasięgu ramienia robota
JAPPRO p,x	porusza się złączowo w kierunku Z narzędzia w pewnej odległości x od punktu p
JUMP p,x	Polecenie specjalne: JUMP do pozycji p, gdzie x jest przegubem lub punktem kartezjańskim, x odpowiada wysokości skoku.
JMOVE p	ruch robota złączowo po pozycji p (joint), gdzie p jest punktem złączowym lub kartezjańskim
LMOVE p	ruch liniowy do punktu p
LAPPRO p,x	porusza się w kierunku Z narzędzia dla danej odległości x od punktu p liniowo

**ASTORINO Instrukcja obsługi**

LDEPART x	przesuwa robota z bieżącej pozycji w określonej odległości x od bieżącego punktu sprzedaży. wzdłuż osi Z narzędzia
POINT p	Tworzy zmienną x punktu
PRINT x	Wypisanie danych/tekstu na terminalu
PULSE x,y	Aktywuje sygnał X (1-58 lub Int. 2001-2016) na czas Y (sek.)
SEND x	Wysyłanie danych do HOST (Serial)
SHIFT(p BY x,y,z)	Tworzy nowy punkt na podstawie przemieszczenia p Przykład: POINT TST = SHIFT(P1 PRZEZ 10,0,0)
SIG(x)	sprawdza stan sygnału x – zwraca wartość TRUE lub FALSE Przykład: IF SIG(2001) == TRUE THEN
SIGNAL x	Aktywuje sygnał X (1-58 lub Int. 2001-2016)
SIGNAL -x	Deaktywuje sygnał X (1-58 lub Int. 2001-2016)
SPEED x	Prędkość robota w % dla polecenia następnego ruchu
SPEED x ALWAYS	prędkość robota w %
SPEED x MM/S	prędkość robota w mm/s (maks. 250 mm/s ) dla polecenia następnego ruchu
SPEED x MM/S ALWAYS	prędkość robota w mm/s (maks. 250 mm/s )
SWAIT x	Wstrzymuje program do wysokiego stanu sygnału X (1-58 lub 2001-2016)
SWAIT -x	Wstrzymuje program do stanu niskiego sygnału X( 1-8 lub 2001-2016 )
TDRAW x,y,z	ruch liniowy względem x,y,z wg TOOL
TOOL p	Wybieranie danych narzędzia z przekształceń punktów p
TOOL x	wybór jednego z systemów TOOL ( x = 1,2,3 )
TWAIT x	wstrzymuje program na x sekund

## ASTORINO Instrukcja obsługi

TYPE x	wypisanie danych/tekstu na terminalu
X = CVPOS	Odczyt danych przenośnika 1 do zmiennej x
X = CVPOS2	Odczyt danych przenośnika 2 do zmiennej X
\$X = RECEIVE	Odczyt danych HOST (Serial) z bufora do zmiennej \$X
Y = VAL(\$X)	Zmienia wartość ciągu na liczbę

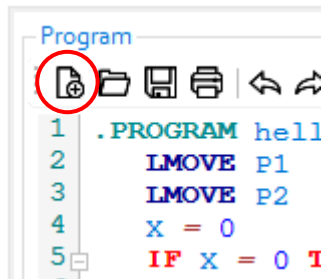
- Wyrażenia warunkowe:
  - IF .... THEN ... ELSE ... END
  - IF .... THEN .... END
  - CASE ... OF .. VALUE ... ANY... END
  -
- Pętle:
  - FOR ... TO ... END
  - DO... UNTIL
  - WHILE ... END
- Wyrażenia i funkcje matematyczne:
  - +, -, \*, /, ^, MOD
  - SIN, COS, ATAN, FABS

## 15.16 Programowanie

### 15.16.1 Tworzenie nowego programu

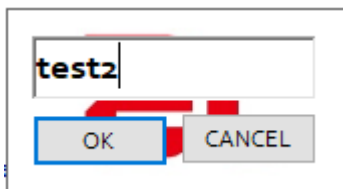
Obszar **Program** znajduje się na karcie **Programs**.

Kliknij ikonę znajdującą się po lewej stronie, aby utworzyć nowy program:

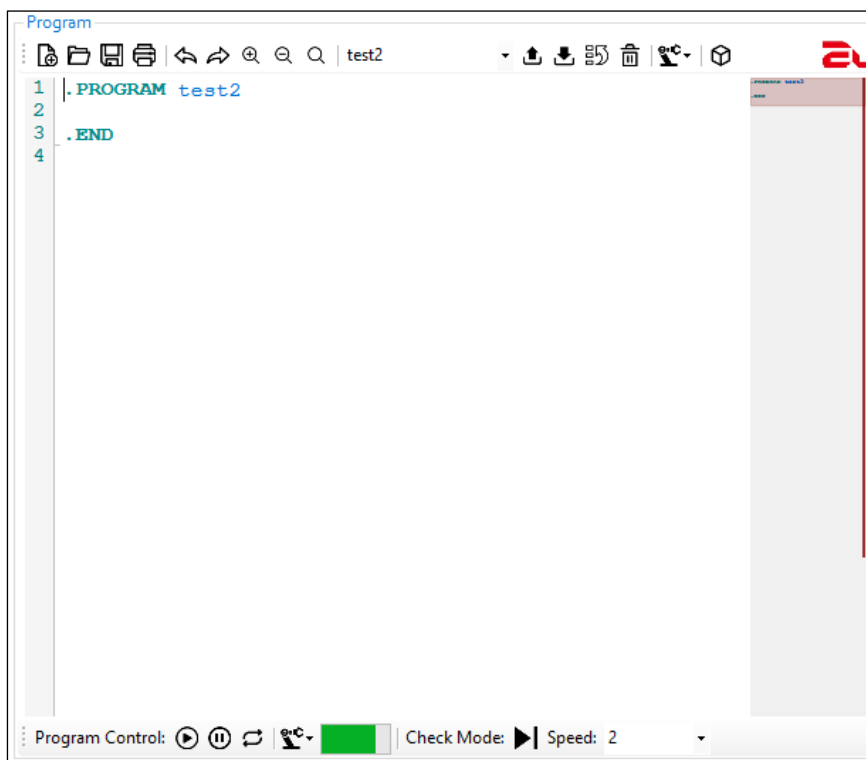


Pojawi się nowe okno.

Wprowadź nazwę programu i kliknij **[OK]**:



Oprogramowanie tworzy nowy szablon programu:



## 15.16.2 Napisz swój własny program

Program

```

1  .PROGRAM test2
2  SPEED 60
3  SPEED 100 MM/S
4  i = 0
5  n = 5
6  FOR i = 0 TO n
7      POINT tst = SHIFT(P1 BY 10*i,0,0)
8      LAPPRO tst, 50
9      LMOVE tst
10     TWAIT 1
11 END
12 .END
13
    
```

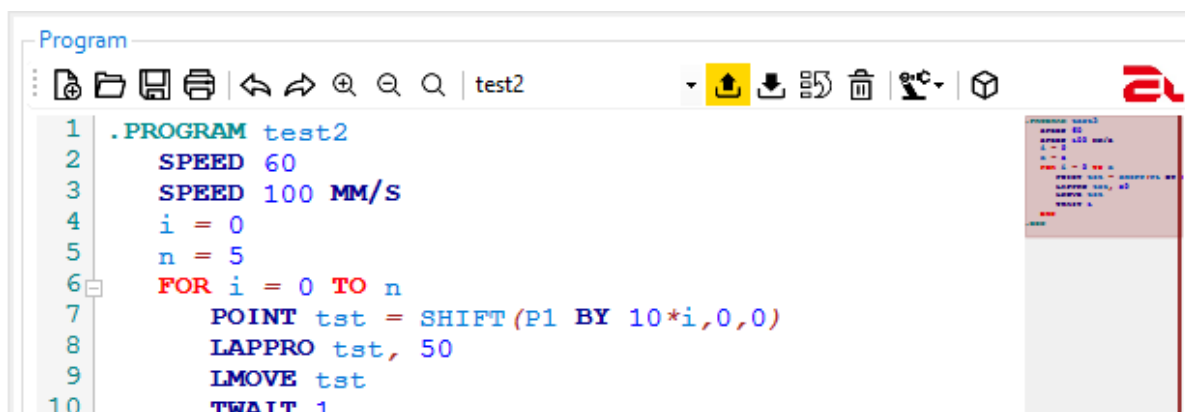
## 15.16.3 Ładowanie programu do robota

Przycisk **[Prześlij]** zacznie migać na żółto. Kliknij na niego, aby wgrać program do pamięci robota:

Program

```

1  .PROGRAM test2
2  SPEED 60
3  SPEED 100 MM/S
4  i = 0
5  n = 5
6  FOR i = 0 TO n
7      POINT tst = SHIFT(P1 BY 10*i,0,0)
8      LAPPRO tst, 50
9      LMOVE tst
10     TWAIT 1
    
```

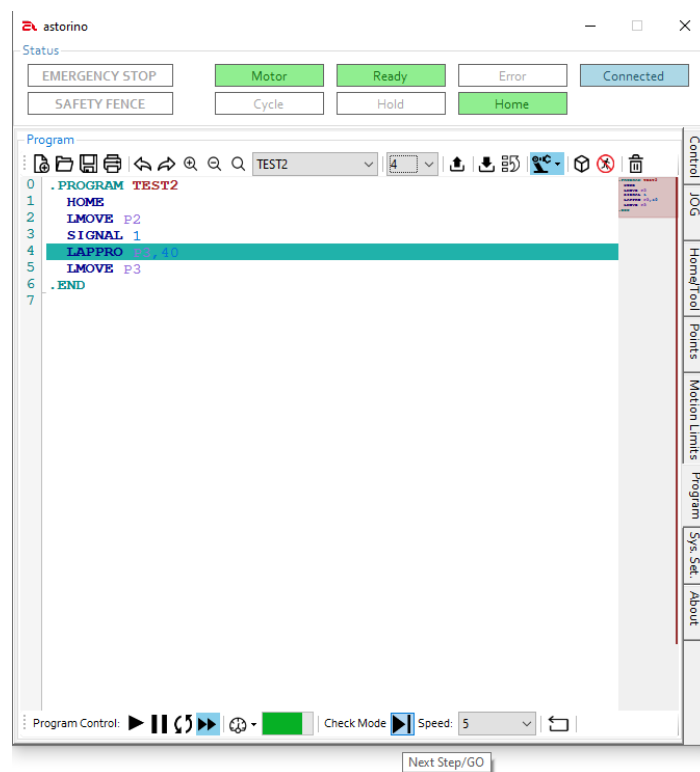


### 15.16.4 Uruchamianie aktualnie wybranej linijki

Aby uruchomić aktualnie zaznaczoną linię, naciśnij i przytrzymaj przycisk Next Step/GO, a jednocześnie naciśnij i przytrzymaj lewy CTRL na klawiaturze. Robot musi być w trybie uczenia.



Zwolnienie przycisku CTRL lub GO spowoduje zatrzymanie robota.



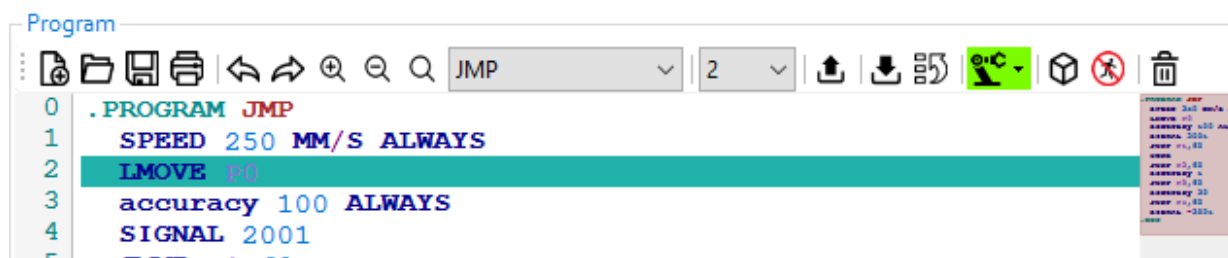
#### **UWAGA**

**Uruchomienie kodu za pomocą przycisku GO spowoduje wykonanie tylko poleceń innych niż sterowania przepływem programu. Polecenia takie jak IF, FOR, WHILE itp. zwrócą błąd.**

### 15.16.5 Uruchamianie programu

#### **UWAGA!**

**Przed uruchomieniem programu upewnij się, że robot w nic nie uderzy. Jeśli nie masz pewności co do napisanego programu, uruchom go najpierw w trybie DryRun!**



Kliknij ikonę Odtwórz na pasku ► sterowania programem [Cycle Start], aby uruchomić program od aktualnie zaznaczonej linijki:

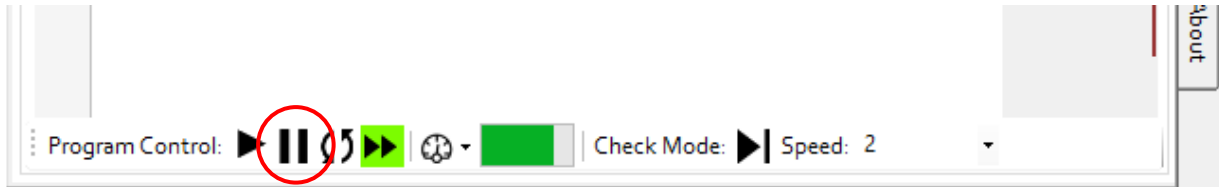


#### **UWAGA!**

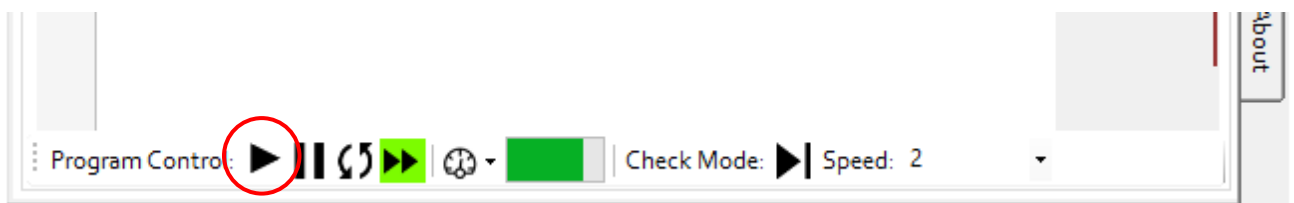
**Uruchomienie programu wewnątrz pętli lub warunku (FOR, IF, WHILE itd.) spowoduje błąd programu.**

## 15.16.6 Zatrzymanie programu

Aby zatrzymać wykonywanie programu należy najpierw wstrzymać pracę robota poprzez przycisk [Wstrzymaj/HOLD]



A następnie po zatrzymaniu robota kliknąć ikonę Odtwórz na pasku ► sterowania programem [Cycle Start], aby zatrzymać program:





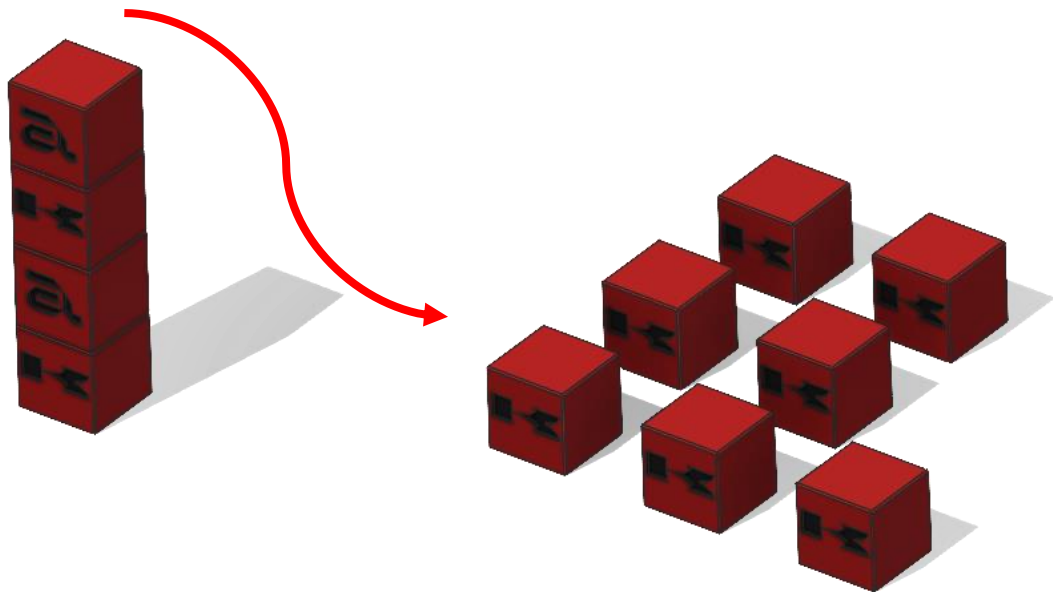
## 16 Przykładowe programy

### 16.1 Pick&Place – przykład paletyzacji

Ten program pobiera kostki z pojedynczej wieży, a następnie umieszcza je według liczby rzędów, liczby kolumn i liczby poziomów.

Użytkownik może dostosować:

- Rozmiar przedmiotu (kostki)
- Odległość między kostkami,
- Liczba rzędów, kolumn i poziomów,



## ASTORINO Instrukcja obsługi

```

. PROGRAM PAL1
;----- Init -----
deltaX = 60 ;distance between workpieces X
deltaY = 60 ;distance between workpieces Y
deltaZ = 30 ;layer height
numLev = 2
numRow = 1
numCol = 2
numPcs = numLev*numCol*numRow ;pieces count
height = 25 ;height of a workpiece (25 mm)
;----- variable init -----
x = 0
y = 0
z = 1
SIGNAL -1
speed 100 mm/s always
POINT place = p2
POINT pick = P1
POINT pick = SHIFT(p1 BY 0,0,numPcs*height)
;P1 on the table, pick shifted by number of pieces in Z
HOME
LAPPRO pick, 40
;----- Pal-----
FOR z = 0 TO (numLev-1)
    FOR y = 0 TO (numRow-1) ; rows in Y
        FOR x = 0 TO (numCol-1) ;col in X
            POINT pick = SHIFT(pick BY 0,0,-height);calc new pick pose
            JAPPRO pick,40
            speed 20 mm/s
            LMOVE pick
            TWAIT 0.5
            SIGNAL 1 ;close the gripper
            TWAIT 0.5
            LDEPART 50
            LMOVE P3
            POINT place = p2
            POINT place = SHIFT(p2 BY deltax*x,deltay*y,deltaz*z)
            LAPPRO place,30
            speed 20 mm/s
            LMOVE place
            TWAIT 0.5
            SIGNAL -1 ;open the gripper
            TWAIT 0.5
            LDEPART 30
            LMOVE P3
        END
    END
END
.END

```

## 16.2 Przykładowy program we/wy

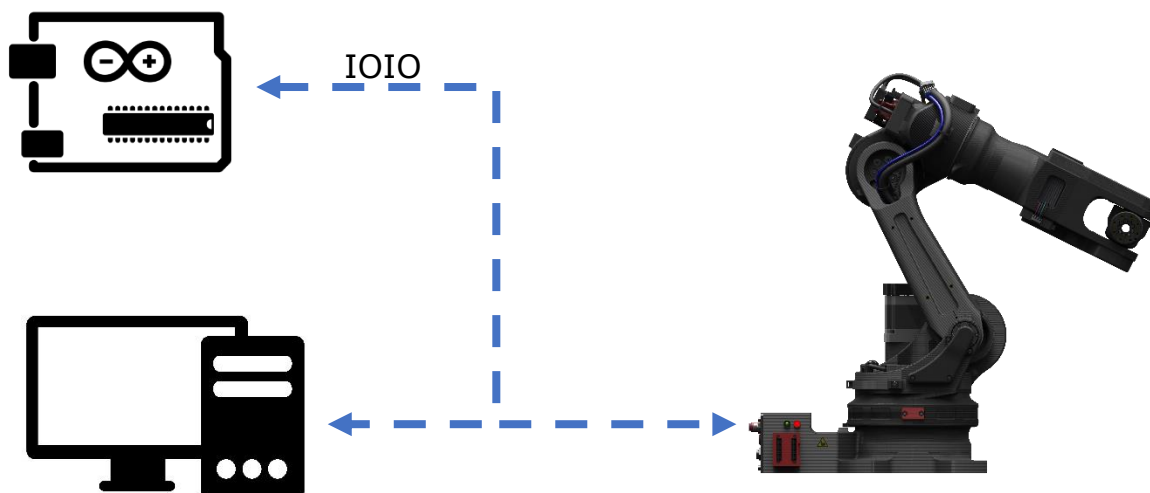
Ten przykładowy program pokazuje, jak używać sygnały na wiele sposobów.

```

PROGRAM IO
; ----- IO example program
; ----- Robot reads and sets IOs
sensor = 1002 ;sets variable
SWAIT 2001 ;wait until internal 1 signal is on
SIGNAL 8 ;sets 8 output HIGH
IF SIG(sensor) == TRUE THEN
    ;checks if sensor(2 input) is high
    SIGNAL 2002 ; sets 2 internal HIGH
ELSE
    IF SIG(1001) == FALSE THEN
        SIGNAL -8 ;sets 1 output LOW
    END
END
END
BITS 1,4 = 12
;changes 12 to 4bit binary and sets that on out puts from 1
data = BITS(1004,4) ;read binary data from inputs
;4 bit from 4th output and changes that to decimal
PRINT data
.END
    
```

## 16.3 Przykładowy program do komunikacji szeregowej

W tym przykładzie pokazano, jak korzystać z komunikacji szeregowej. Program może wymieniać dane pomiędzy robotem Astorino, a komputerem PC (np. Matlab lub SerialTerminal) lub mikrokontrolerem (np. Arduino lub ESP32).



**Wtyczka M8 - (Piny: 1-GND, 2 - 5V [Wizja], 3-TX, 4- RX)**

## ASTORINO Instrukcja obsługi

```
.PROGRAM SERIAL
; ----- Serial communication example program
; ----- Robot command frame form Serial Port
; ----- frames: P/ or L/x/y/z/
; ----- From X,Y,Z point is created
; ----- Sends current location if frame is P/
SPEED 150 MM/S ALWAYS
$$_FRAME = "XYZ"
$$_FRAME2 = "JT"
WHILE EXISTCOM == FALSE DO
    TWAIT 0.1
END
$TEMP = RECEIVE
$COMMAND = $DECODE($TEMP, "/")
PRINT $COMMAND
;RECEIVE DATA FROM SERIAL AND CREATE A POINT
IF $COMMAND == "L" THEN
    $VAL1 = $DECODE($TEMP, "/")
    $VAL2 = $DECODE($TEMP, "/")
    $VAL3 = $DECODE($TEMP, "/")
    DATAX = VAL($VAL1)
    DATAY = VAL($VAL2)
    DATAZ = VAL($VAL3)
    POINT TEST = TRANS(DATAX, DATAY, DATAZ, 0, 0, 0)
    POINT/OAT TEST = P0
    LMOVE TEST
    SEND "OK"
END
;SEND CURRENT LOCATION TO SERIAL PORT
IF $COMMAND == "P" THEN
    HERE TEMP
    HERE #TEMP
    DECOMPOSE TAB[0] = TEMP
    DECOMPOSE TAB2[0] = #TEMP
    FOR I = 0 TO 5
        TAB2[I] = TAB2[I]*180/PI
        $$_FRAME = $$_FRAME + $ENCODE(TAB[I]) + "/"
        $$_FRAME2 = $$_FRAME2 + $ENCODE(TAB2[I]) + "/"
    END
    SEND $$_FRAME
    SEND $$_FRAME2
END
.END
```

### UWAGA

**Komunikacja szeregową działa na napięciu 3.3V, proszę używać elektroniki kompatybilnej z napięciem 3.3V lub konwerterów poziomów napięć.**

**Napięcie 5V może uszkodzić główny układ CPU!**

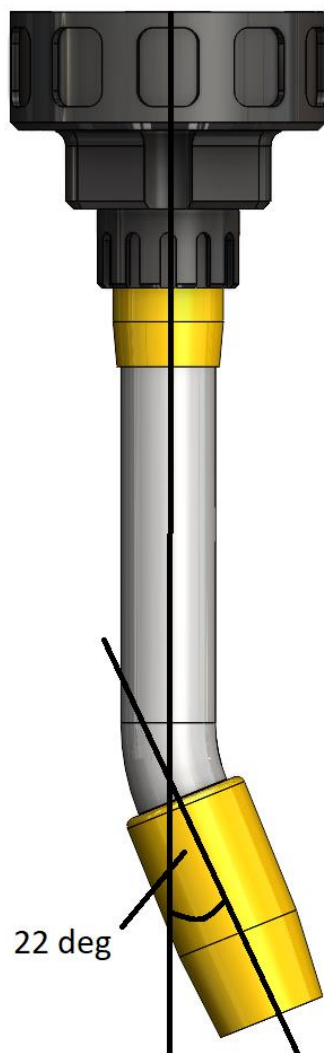
## 17 Dane narzędzia

### 17.1 Dane narzędzia ze znanych wymiarów

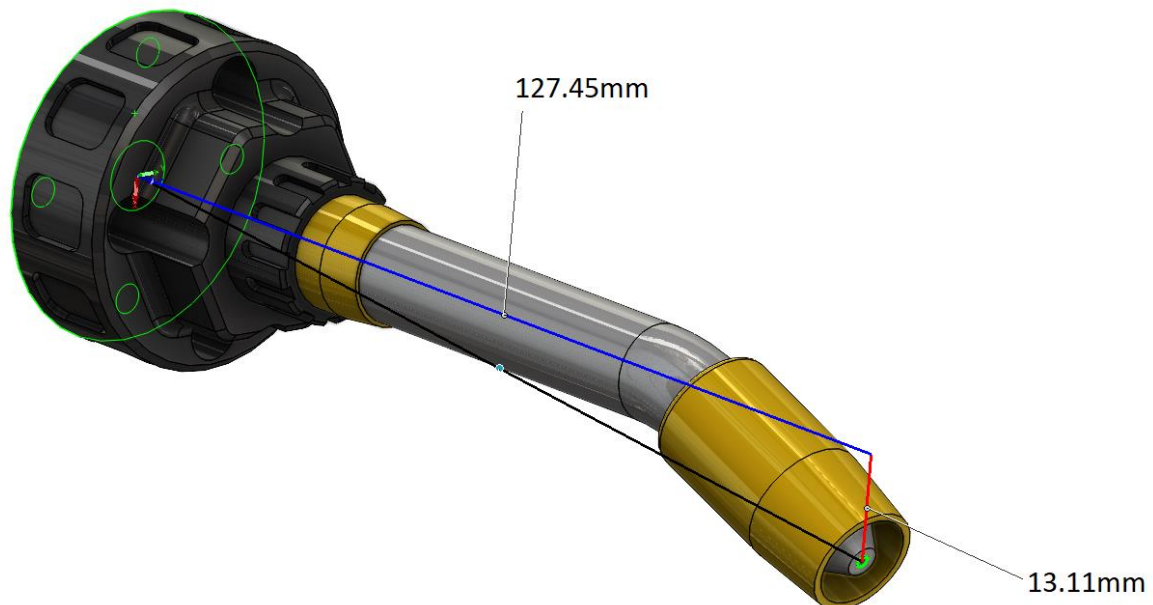
Wymiary narzędzia (TOOL DATA) podawane są w podstawowym układzie współrzędnych (BASE) robota.

Wprowadź wymiary narzędzia z istniejących danych CAD lub zmierz je samodzielnie.

Poniższy przykład palnika pokazuje, w jaki sposób można samodzielnie określić takie dane narzędzia:



Palnik jest ustawiony pod kątem 22 stopni.



W tym przypadku długość palnika od powierzchni kołnierza (oś 6) do końcówki dyszy wynosi 127,45 mm w kierunku Z.

(Uwaga! Pracujesz teraz ze współrzędnymi narzędzia)

Palnik powoduje przesunięcie w kierunku Y o 13,11 mm. Punktem odniesienia jest zawsze powierzchnia kołnierza narzędzia. Układ współrzędnych ma swój początek w środku powierzchni kołnierza, gdzie kierunek Z = kierunek ciągu narzędzia w przestrzeni.

W tym przykładzie należy wprowadzić następujące wartości:

X [mm]	0. 0	
T [mm]	13. 11	
Z[mm]	127. 45	
Rx: -22,0	[O: 90. 0]	Rx to obrót wokół osi X
Ry: 0.0	[O: 22. 0]	Ry jest obrotem wokół osi Y
Rz: 0,0	[T: -90. 0]	Rz jest obrotem wokół osi Z

## 17.2 Automatyczna konfiguracja narzędzia (współrzędne)

W tym rozdziale opisano procedury działania automatycznej rejestracji wartości współrzędnych narzędzia.

### UWAGA

**Automatyczna konfiguracja narzędzi jest rodzajem operacji ruchu robota w trybie TEACH. Jego zastosowanie jest ograniczone do personelu, który ukończył specjalne szkolenie i posiada kwalifikacje do nauczania lub nadzorowania pracy robota.**

### 17.2.1 Przegląd funkcji automatycznej konfiguracji narzędzi

Podczas obsługi robota na flanszy nadgarstka można zamontować różnorodne narzędzia o różnych kształtach (chwytak itp.). Jeśli dane narzędzia nie są wzmierzone prawidłowo, trajektoria ruchu robota może odbiegać od nauczonej ścieżki, a wszelkie błędy lub awarie mogą się powiększyć.

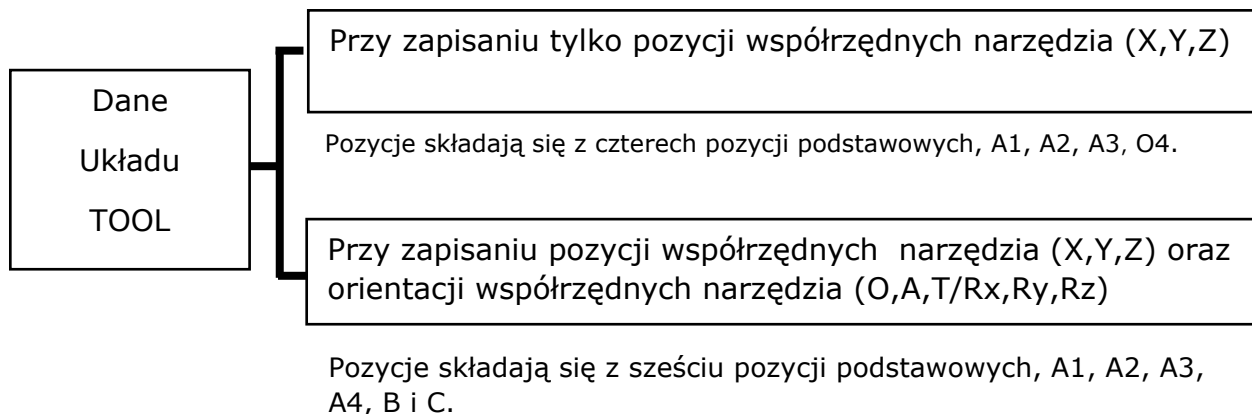
Innymi słowy, dane narzędzia są niezbędne do prawidłowej obsługi robota. Ogólnie rzecz biorąc, dane narzędzia są wprowadzane za pomocą wartości liczbowych lub obliczane, należy jednak pamiętać, pomiar położenia i orientacji współrzędnych narzędzia może nie być dokładna lub może wymagać długiego czasu do nabycia doświadczenia w uczeniu punktów.

Funkcja ta umożliwia automatyczną konfigurację wartości transformacji narzędzia poprzez uczenie kilku punktów w przestrzeni bez konieczności wprowadzania wartości danych narzędzia za pomocą danych numerycznych. Do tego potrzebny jest przyrząd pomiarowy z ostrym końcem, na przykład duży wkręt i spiczasty stożek na narzędziu.



## 17.2.2 Dane wymagane do automatycznej konfiguracji współrzędnych narzędzia

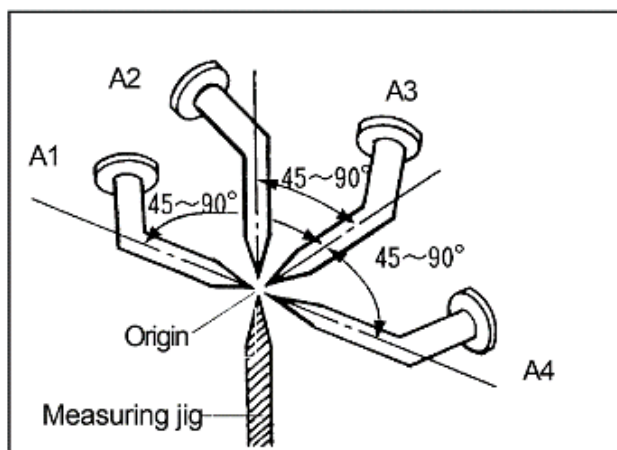
Podczas korzystania z funkcji automatycznej konfiguracji narzędzia przechowywany jest następujący zestaw danych ułożenia zgodnie ze stanem danych narzędzia. Pomiar danych pozy odbywa się poprzez celowanie w jeden punkt nauczania z 4 lub 6 różnych pozycji narzędzi, jak opisano poniżej.



## 17.2.3 Nauczanie czterech podstawowych pozycji

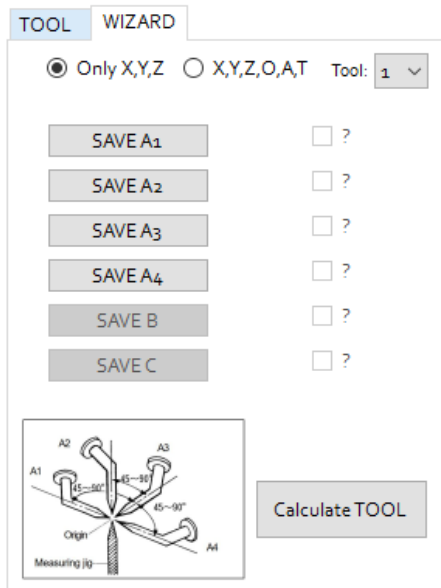
Czteropunktowa metoda obliczeń TCP umożliwia znalezienie przesunięć danych TOOL w wymiarach X, Y, Z.

Jak pokazano na poniższym rysunku, naucz 4 pozycje podstawowe (A1, A2, A3, A4) z tymi samymi danymi pozycji, ale z różnymi danymi orientacyjnymi, celując w ten sam punkt końcówki na przyrządzie pomiarowym. Upewnij się, że kąty między poszczególnymi orientacjami mieszczą się w zakresie od 45° do 90°. Powierzchnia czołowa flanszy nadgarstka powinna mieć inną płaszczyznę dla każdej orientacji punktu. Naucz każdą pozycję bazową tak, aby współrzędne narzędzia i początki przyrządu pomiarowego stykały się ze sobą.





## ASTORINO Instrukcja obsługi

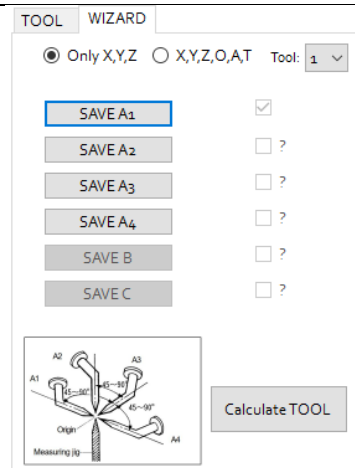


Wybierz [Only X, Y,Z] w zakładce HOME/Tool

Wybierz z listy numer narzędzia, którego chcesz uczyć. Możesz wybrać 1,2 lub 3

Przełącz robota w tryb uczenia i przejdź do pozycji jak poniżej (jest to przykład, rzeczywiste pozycje mogą być inne). Po osiągnięciu pozycji naciśnij SAVE Ax, gdzie x wynosi 1,2,3 lub 4

### Naucz A1



# ASTORINO Instrukcja obsługi

## Naucz A2



TOOL WIZARD

Only X,Y,Z  X,Y,Z,O,A,T Tool: 1

SAVE A1

SAVE A2

SAVE A3  ?

SAVE A4  ?

SAVE B  ?

SAVE C  ?

Calculate TOOL

## Naucz A3



TOOL WIZARD

Only X,Y,Z  X,Y,Z,O,A,T Tool: 1

SAVE A1

SAVE A2

SAVE A3

SAVE A4  ?

SAVE B  ?

SAVE C  ?

Calculate TOOL

## Naucz A4



TOOL WIZARD

Only X,Y,Z  X,Y,Z,O,A,T Tool: 1

SAVE A1

SAVE A2

SAVE A3

SAVE A4

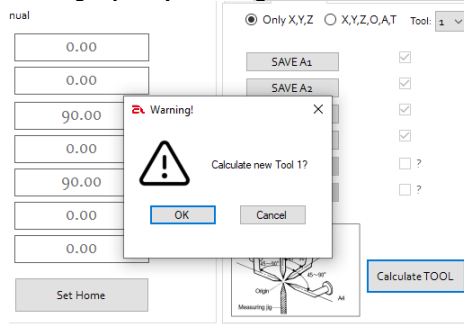
SAVE B  ?

SAVE C  ?

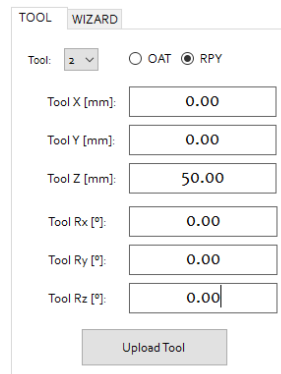
Calculate TOOL

## ASTORINO Instrukcja obsługi

Naciśnij przycisk [Calculate TOOL]

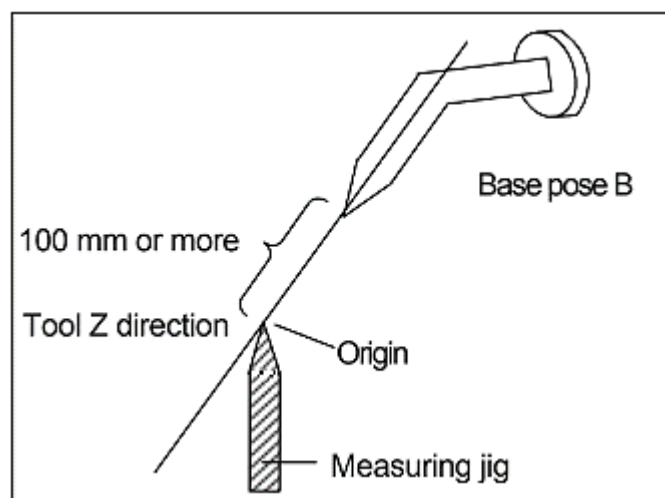


Nowe dane obliczeniowe narzędzia zostaną zapisane na karcie SD i wyświetlone na karcie TOOL.



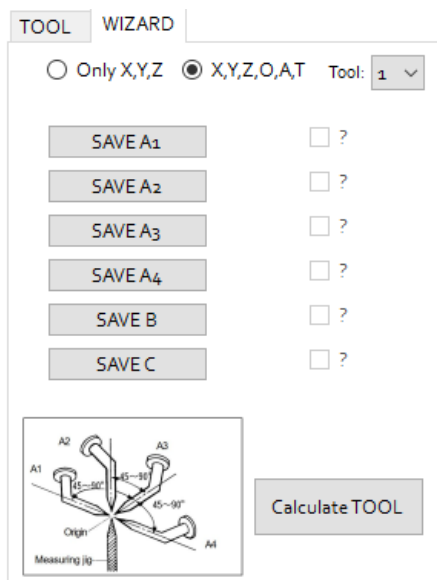
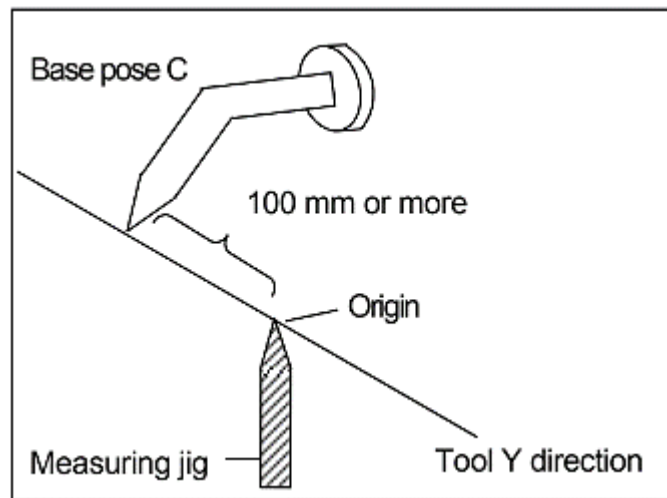
### 17.2.4 Nauczanie sześciu podstawowych pozycji

W przypadku pozycji podstawowej B naucz ją tak, aby stykała się między początkiem przyrządu pomiarowego a pozycją oddaloną o 100 mm lub więcej od TCP (punkt środkowy narzędzia) w pożądanym kierunku -Z narzędzia, jak pokazano poniżej.



## ASTORINO Instrukcja obsługi

W przypadku podstawowej pozycji C naucz ją tak, aby stykała się między początkiem przyrządu pomiarowego, a pozycją oddaloną o 100 mm lub więcej od TCP w pożądanym kierunku +Y narzędzia, jak pokazano poniżej.



Wybierz [X,Y,Z,O,A,T] w zakładce HOME/Tool (Narzędzie)

Wybierz z listy numer narzędzia, którego chcesz uczyć. Możesz wybrać 1,2 lub 3

Przełącz robota w tryb uczenia i przejdź do pozycji jak poniżej (jest to przykład, rzeczywiste pozycje mogą być inne). Po osiągnięciu pozycji naciśnij SAVE Ax, gdzie x wynosi 1,2,3 lub 4

# ASTORINO Instrukcja obsługi

## Naucz A1



The screenshot shows the 'TOOL WIZARD' interface for the 'Naucz A1' step. The robot arm is positioned to measure a green cone. The 'TOOL WIZARD' panel includes the following elements:

- Buttons: **SAVE A1** (highlighted), SAVE A2, SAVE A3, SAVE A4, SAVE B, SAVE C.
- Checkboxes:  for SAVE A1, and  with a question mark for SAVE A2, SAVE A3, SAVE A4, SAVE B, and SAVE C.
- Radio buttons:  Only X,Y,Z and  X,Y,Z,O,A,T.
- Tool selection: Tool: 1 (dropdown).
- Diagram: A schematic of the robot arm joints (A1, A2, A3, A4) and the measuring tip.
- Button: Calculate TOOL.

## Naucz A2



The screenshot shows the 'TOOL WIZARD' interface for the 'Naucz A2' step. The robot arm is positioned to measure a green cone. The 'TOOL WIZARD' panel includes the following elements:

- Buttons: SAVE A1, **SAVE A2** (highlighted), SAVE A3, SAVE A4, SAVE B, SAVE C.
- Checkboxes:  for SAVE A1,  for SAVE A2, and  with a question mark for SAVE A3, SAVE A4, SAVE B, and SAVE C.
- Radio buttons:  Only X,Y,Z and  X,Y,Z,O,A,T.
- Tool selection: Tool: 1 (dropdown).
- Diagram: A schematic of the robot arm joints (A1, A2, A3, A4) and the measuring tip.
- Button: Calculate TOOL.

## Naucz A3

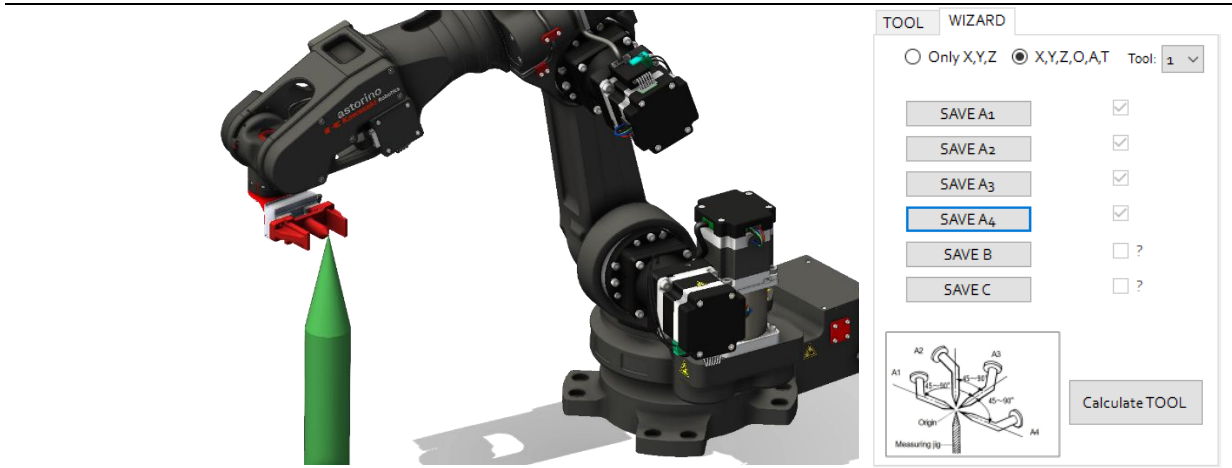


The screenshot shows the 'TOOL WIZARD' interface for the 'Naucz A3' step. The robot arm is positioned to measure a green cone. The 'TOOL WIZARD' panel includes the following elements:

- Buttons: SAVE A1, SAVE A2, **SAVE A3** (highlighted), SAVE A4, SAVE B, SAVE C.
- Checkboxes:  for SAVE A1,  for SAVE A2,  for SAVE A3, and  with a question mark for SAVE A4, SAVE B, and SAVE C.
- Radio buttons:  Only X,Y,Z and  X,Y,Z,O,A,T.
- Tool selection: Tool: 1 (dropdown).
- Diagram: A schematic of the robot arm joints (A1, A2, A3, A4) and the measuring tip.
- Button: Calculate TOOL.

# ASTORINO Instrukcja obsługi

## Naucz A4



**TOOL WIZARD**

Only X,Y,Z  X,Y,Z,O,A,T Tool: 1

SAVE A1

SAVE A2

SAVE A3

**SAVE A4**

SAVE B  ?

SAVE C  ?

Calculate TOOL

## Naucz B



**TOOL WIZARD**

Only X,Y,Z  X,Y,Z,O,A,T Tool: 1

SAVE A1

SAVE A2

SAVE A3

SAVE A4

**SAVE B**

SAVE C  ?

Calculate TOOL

## Naucz C



**TOOL WIZARD**

Only X,Y,Z  X,Y,Z,O,A,T Tool: 1

SAVE A1

SAVE A2

SAVE A3

SAVE A4

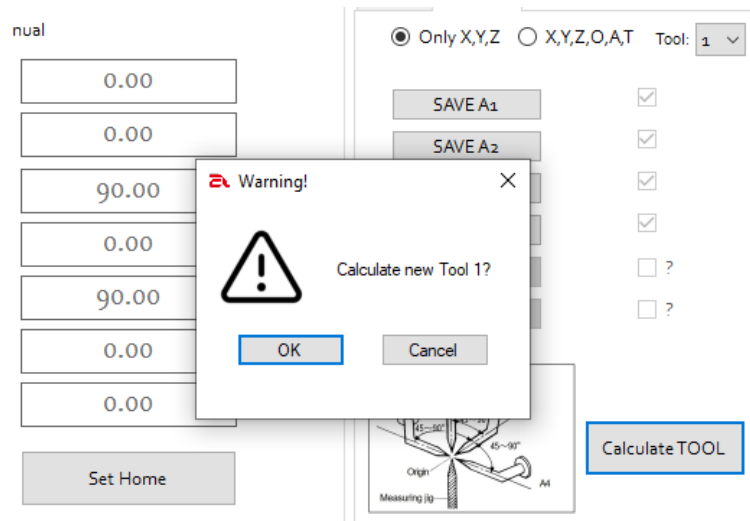
SAVE B

**SAVE C**

Calculate TOOL

## ASTORINO Instrukcja obsługi

Naciśnij przycisk [Oblicz NARZĘDZIE]



Nowe dane obliczeniowe narzędzia zostaną zapisane na karcie SD i wyświetlone na karcie TOOL.

**TOOL WIZARD**

Tool: 1  OAT  RPY

Tool X [mm]: -0.24

Tool Y [mm]: 13.43

Tool Z [mm]: 127.11

Tool Rx [°]: 11.34

Tool Ry [°]: 17.44

Tool Rz [°]: 124.02

Upload Tool

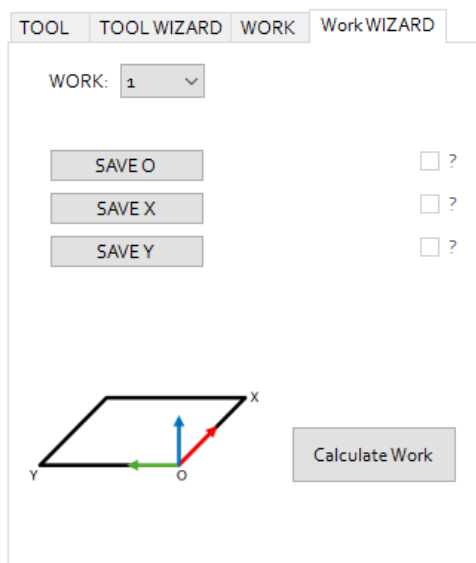
## 18 Automatyczna konfiguracja układu WORK (WSPÓŁRZĘDNE)

W tym rozdziale opisano procedury działania automatycznej rejestracji wartości współrzędnych układu WORK

### UWAGA

**Automatyczna konfiguracja układu WORK jest rodzajem operacji ruchu robota w trybie TEACH. Jego zastosowanie jest ograniczone do personelu, który ukończył specjalne szkolenie i posiada kwalifikacje do nauczania lub nadzorowania pracy robota.**

### 18.1.1 Uczenie trzech podstawowych punktów



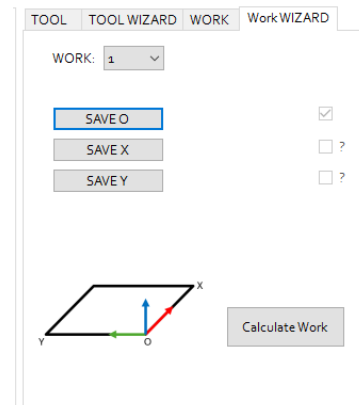
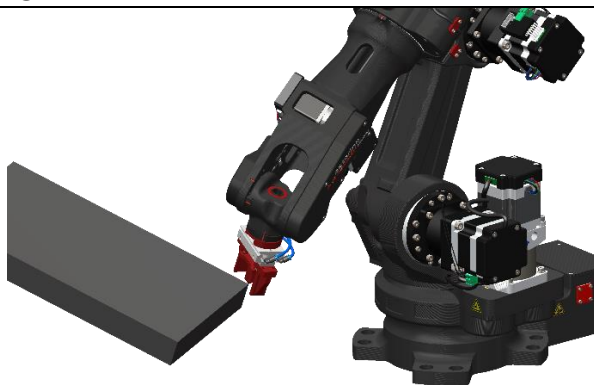
Wybierz numer układu WORK, który będzie obliczony. 1 lub 2 może zostać wybrane.

Przełącz robota w tryb uczenia i przejedź do pozycji jak poniżej (to jest przykład, rzeczywiste pozycje mogą być inne). Po osiągnięciu każdej pozycji naciśnij przycisk ZAPISZ O, X lub Y. Punkt O definiuje początek układu współrzędnych, punkt X definiuje kierunek nowej osi X, punkt Y definiuje kierunek nowej osi Y.

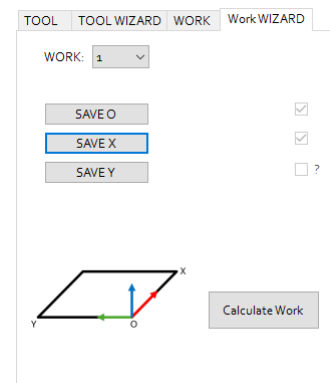
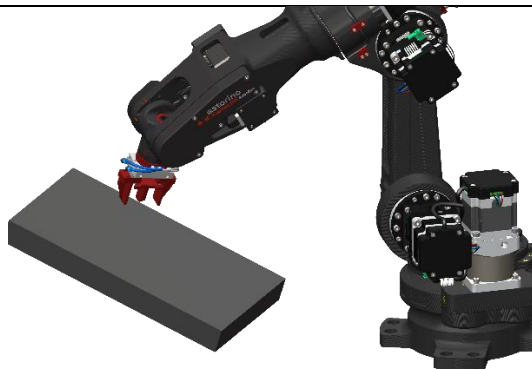


## ASTORINO Instrukcja obsługi

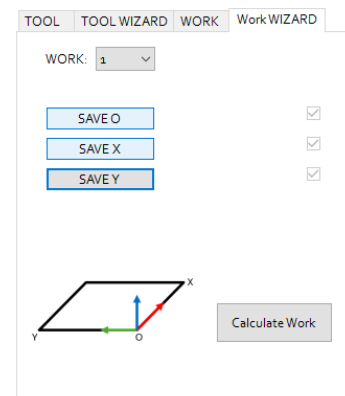
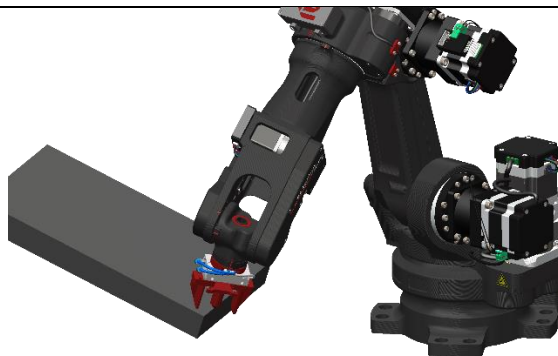
### Naucz O.



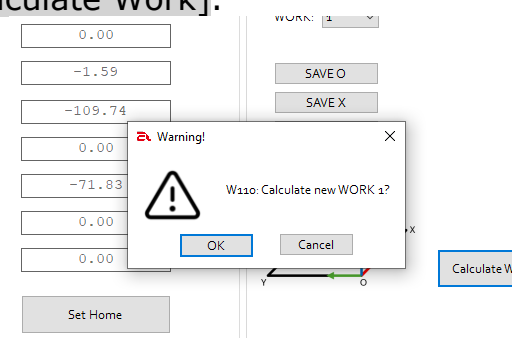
### Naucz X.



### Naucz Y.

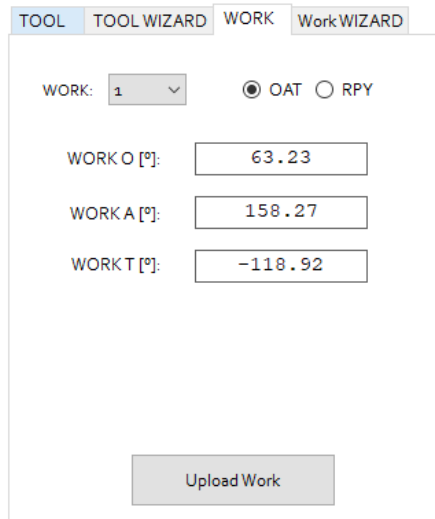


### Naciśnij przycisk [Calculate Work].



## ASTORINO Instrukcja obsługi

Nowe obliczone dane układu WORK zostaną zapisane na karcie SD i wyświetlone w zakładce WORK.



TOOL TOOL WIZARD WORK Work WIZARD

WORK: 1  OAT  RPY

WORK O [°]: 63.23

WORK A [°]: 158.27

WORK T [°]: -118.92

Upload Work

## 19 Auto-kalibracja detekcji kolizji

### [UWAGA]

Kalibracja jest wykonywana dla określonego programu, zmiana programu może wymagać powtórzenia tej procedury!





Aby automatycznie skalibrować progi wykrywania kolizji, przejdź do Kolizja zmień poziom użytkownika na 3.

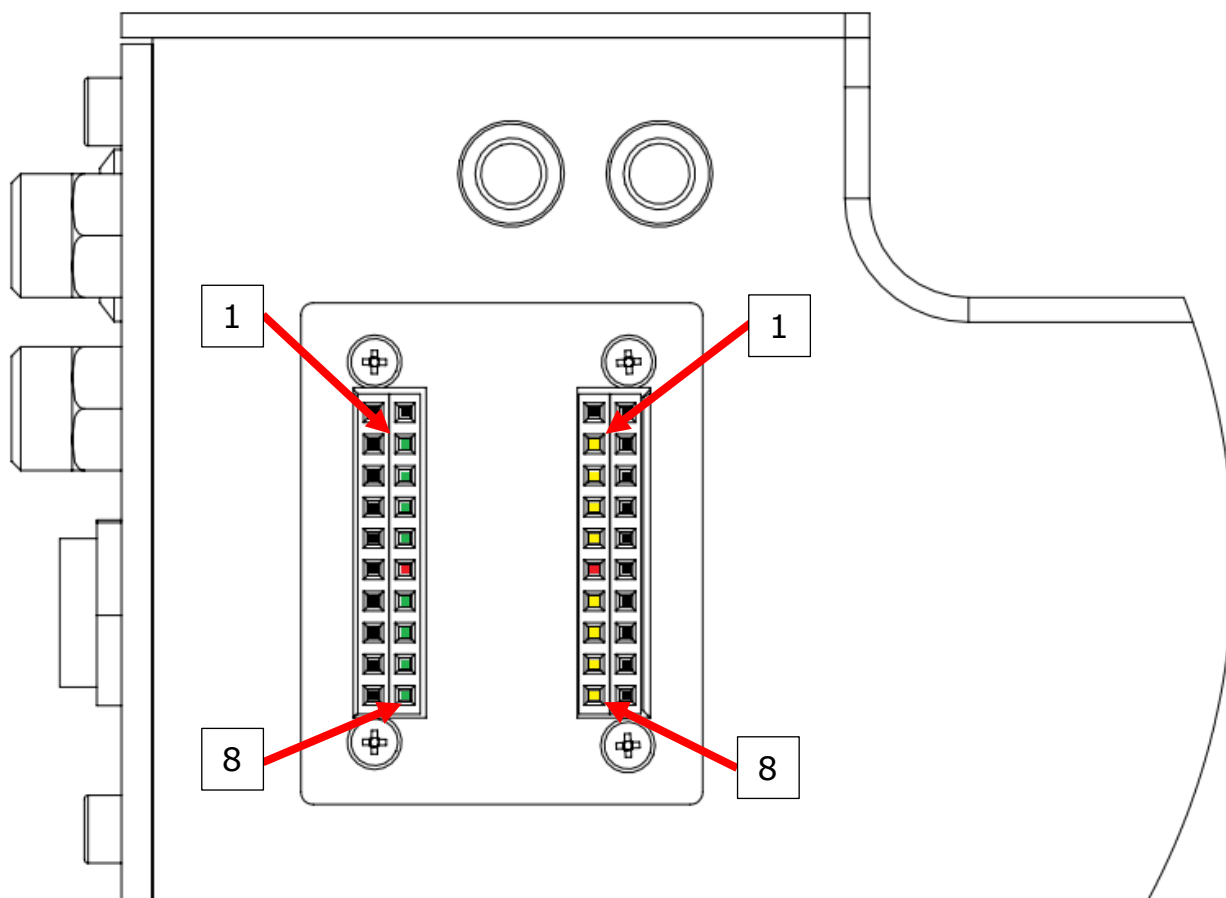
Postępuj zgodnie z poniższymi krokami:

- Przełącz robota w tryb Teach lub Repeat (kalibracja zostanie wykonana dla ostatnio wybranego trybu),
- Wybierz program, dla którego chcesz skalibrować czujnik,
- Przejdź do karty Collision detection, a następnie kliknij [Calibration ON/OFF]
- Uruchom wybrany program na parę cykli,
- Zatrzymaj program (Cycle off),
- Przejdź do zakładki Collision detection i naciśnij [Calibration ON/OFF],
- Nowe progi zostaną zapisane w pamięci robota.

## 20 Wejścia/wyjścia – 3,3V

Astorino posiada 8 wejść i 8 wyjść opartych na napięciu 3,3V DC.

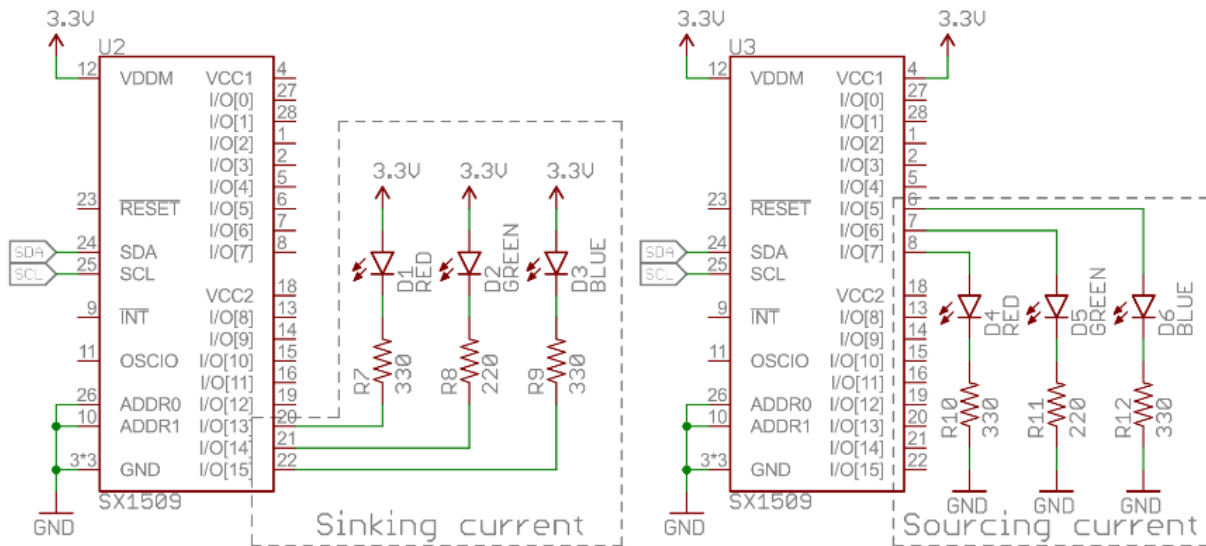
Kolory	Funkcja
	Wyjście
	Wejście
	3,3 V/DC
	0V (GND)



## ASTORINO Instrukcja obsługi

System normalnie pracuje w trybie przełączania PNP. PNP oznacza przełączanie dodatnie (stosowane głównie w Europie i Ameryce Północnej).

W związku z tym moduł przełącza potencjał dodatni na swoje wyjście.



Operację można zmienić na NPN, używając następujących poleceń w terminalu:

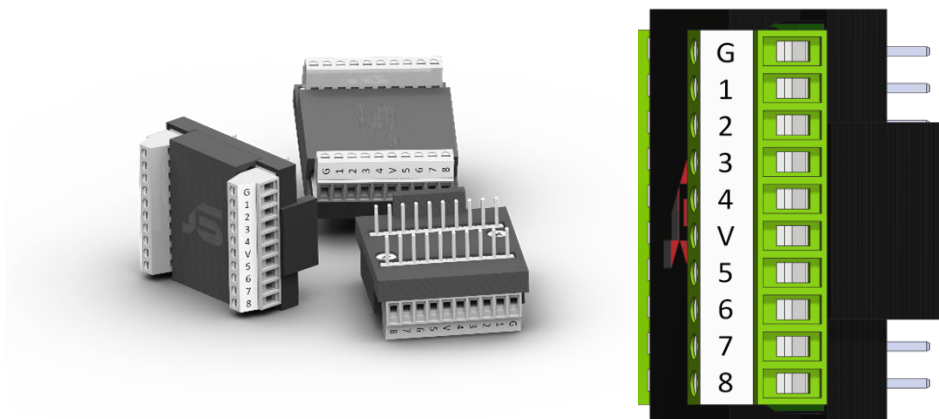
- Z\_OUTSOURCE 1 – PNP
- Z\_OUTSOURCE 0 – NPN
- Z\_INPULL 1 - aktywuje podciągnięcie wejść do 3,3V
- Z\_INPULL 0 - deaktywuje podciągnięcie wejść do 3,3V

### **! UWAGA!**

**Każde WYJŚCIE dostarcza 8 mA prądu. Prosimy nie przekraczać limitu, ponieważ może to uszkodzić płytę główną.**

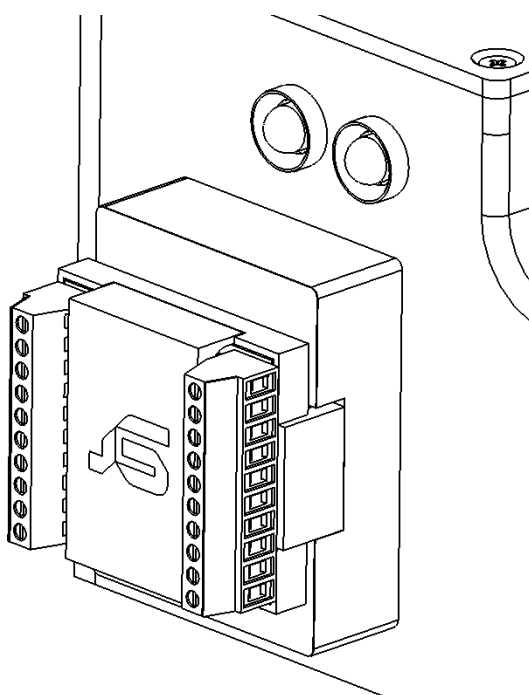
## 20.1 Adapter I/O 3.3V

Adapter IO 3.3V jest opcja i jest sprzedawany oddzielnie. Adapter umożliwia w łatwy sposób podłączenia zewnętrznych urządzeń poprzez złącza śrubowe.



### 20.1.1 Adapter I/O 3.3V – montaż

Adapter wpina się do standardowych Wejść/Wyjść w podstawie robota.



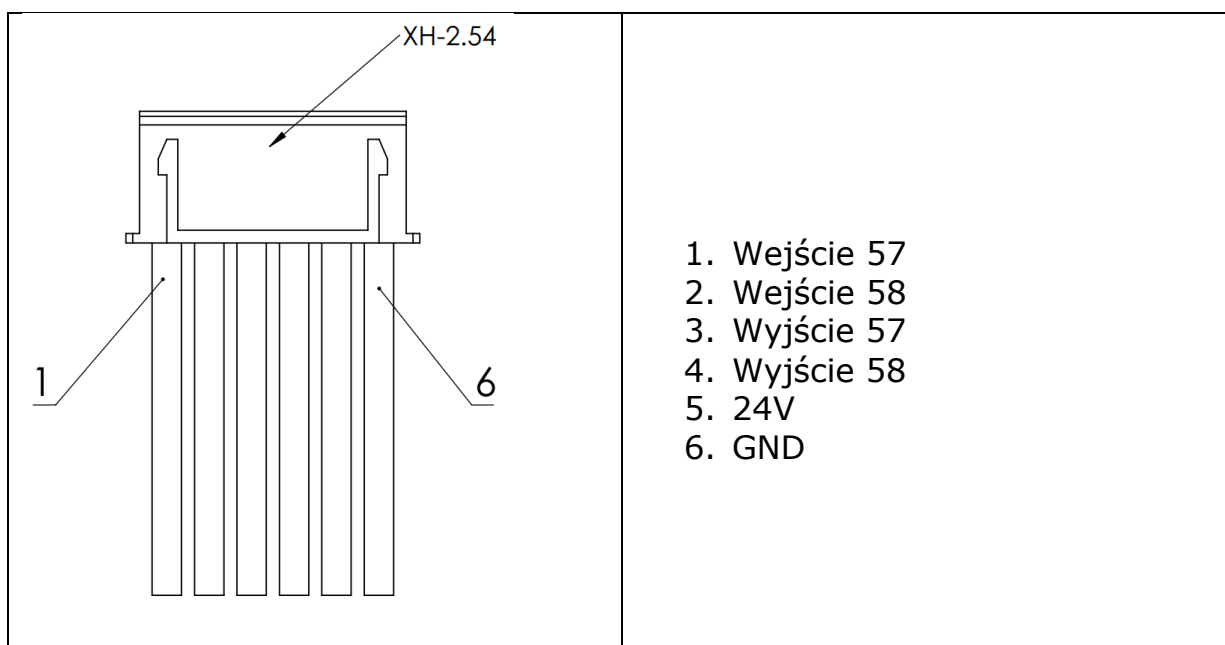
## 21 Wyjścia oraz wejścia na ramieniu robota

Wersja B robota astorino wyposażona jest w dwa wejścia 24V i dwa wyjścia 24V (PNP) na ramieniu JT3.

W oprogramowaniu astorino i systemie AS ich numery to 57 i 58 dla wyjść oraz 1057, 1058 dla wejść.

Zastosowane złącze to XH-2.54 6 pin żeńskie.

Proszę zapoznać się z poniższą tabelą w celu podłączenia WEJŚĆ/WYJŚĆ

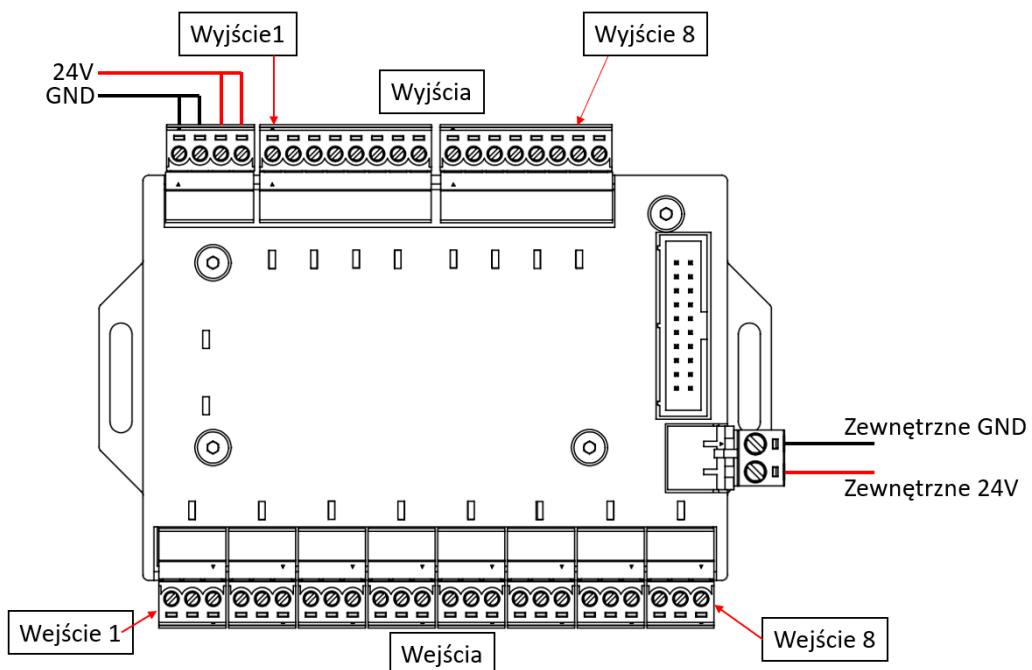
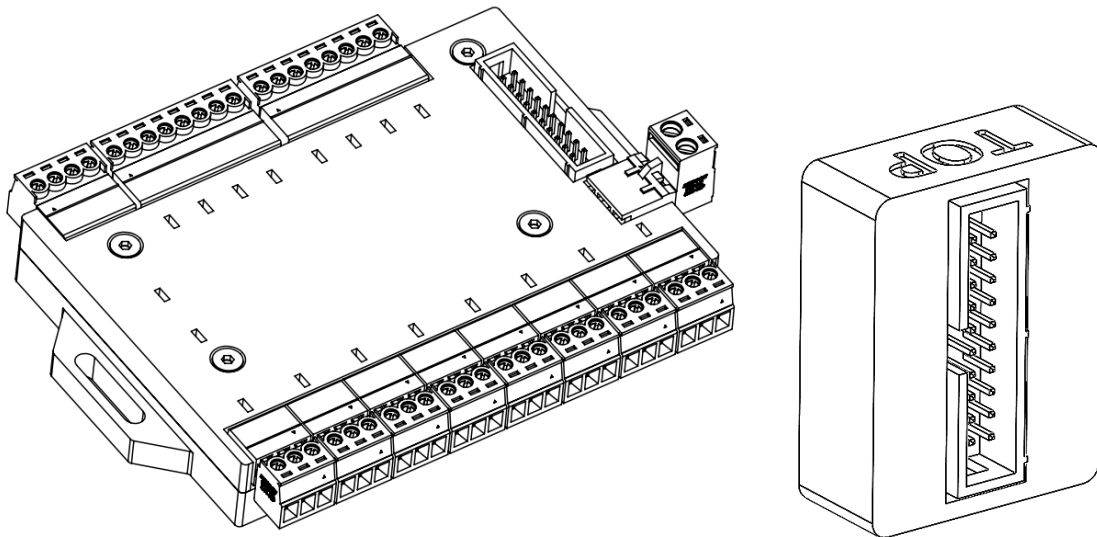


 **UWAGA!**

**Każde WYJŚCIE dostarcza 300 mA prądu. Prosimy nie przekraczać limitu, ponieważ może to uszkodzić płytę główną.**

## 22 Moduł I/O 24V

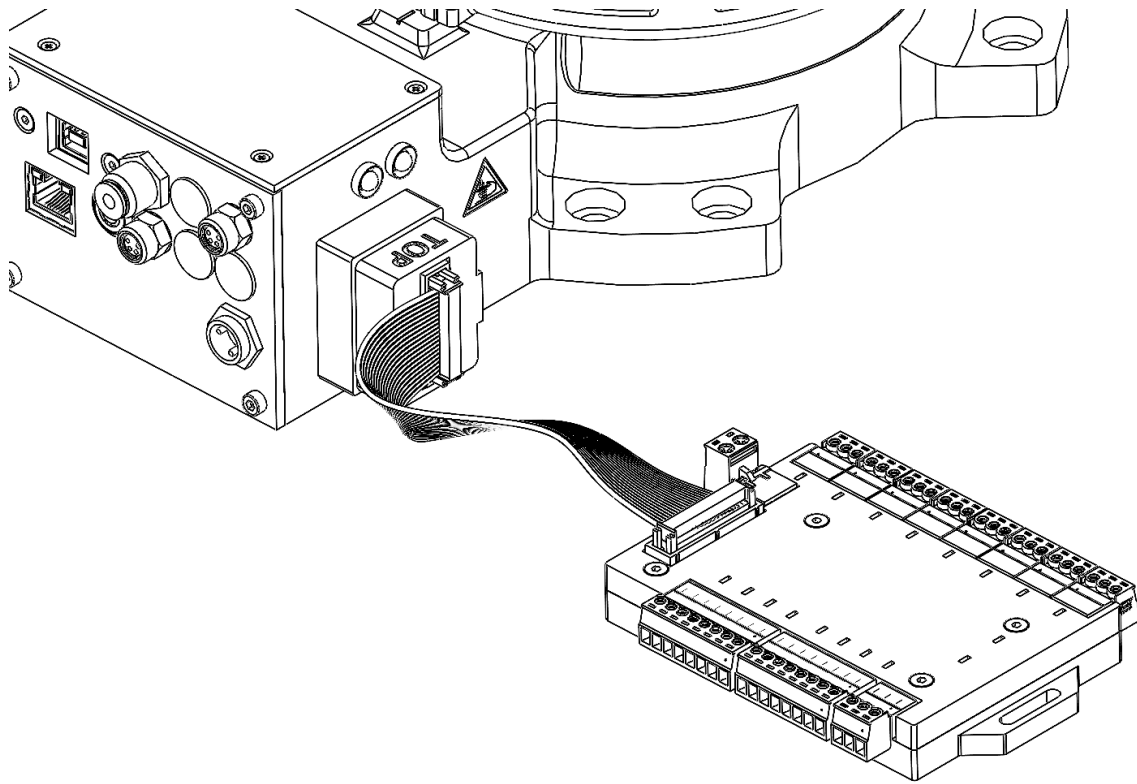
Moduł I/O 24V jest dostępny jako opcja i jest sprzedawany niezależnie od robota Astorino. **Moduł nie rozszerza ilości Wejść/Wyjść, jest nakładką na standardowe IO 3.3V**



Można podłączyć łącznie 8 wejść 24V i 8 wyjść 24V. Każde wyjście zapewnia prąd 300mA (ok. 7,2W).

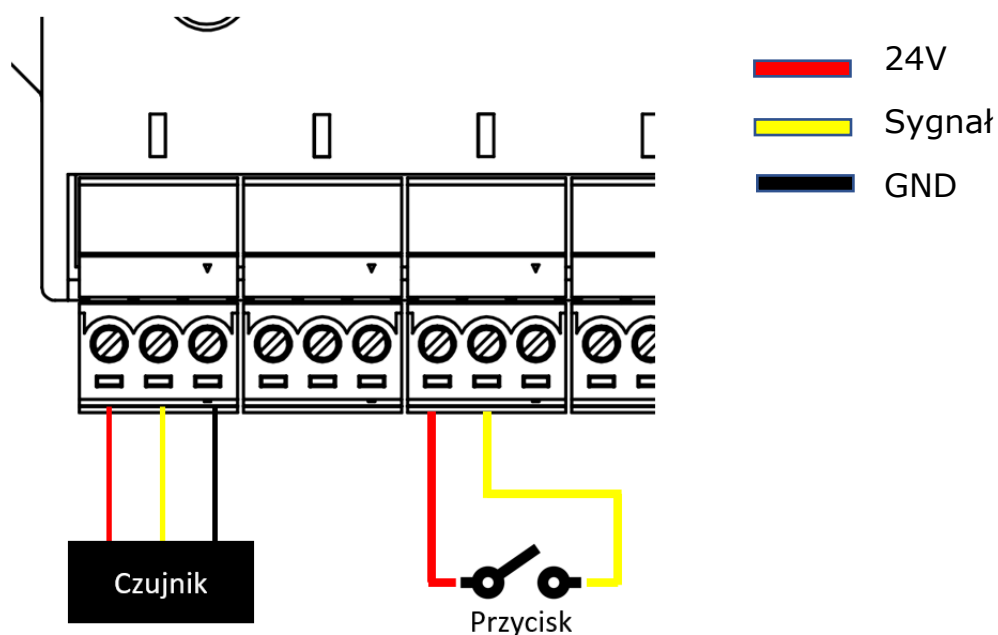
## 22.1 Podłączenie do robota

Moduł podłącza się poprzez dostarczony 30cm przewód wstążkowy.



## 22.2 Podłączanie wejść

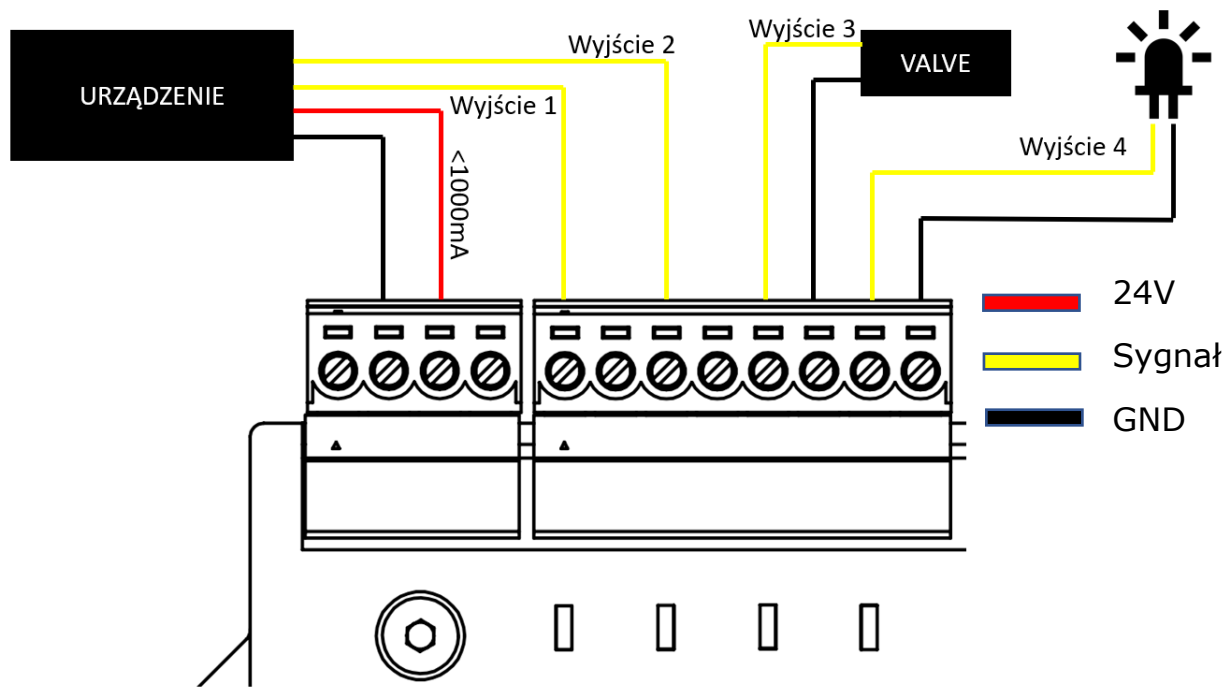
Przykładowe podłączenie urządzeń zewnętrznych do wejść robota





## 22.3 Podłączanie wyjść

Przykładowe podłączenie urządzeń zewnętrznych do wyjść robota.



## 23 MODBUS TCP

Modbus to protokół transmisji danych pierwotnie opublikowany przez Modicon (obecnie Schneider Electric) w 1979 roku do użytku z programowalnymi sterownikami logicznymi (PLC). Modbus stał się de facto standardowym protokołem komunikacyjnym i jest powszechnie dostępnym środkiem podłączenia przemysłowych urządzeń elektronicznych.



### UWAGA!

**Jeżeli astorino działa jako urządzenie typu klient w sieci Modbus to rejestry nie są odświeżane w czasie trwania ruchu robota!**

### 23.1 Tryby pracy sieci Modbus

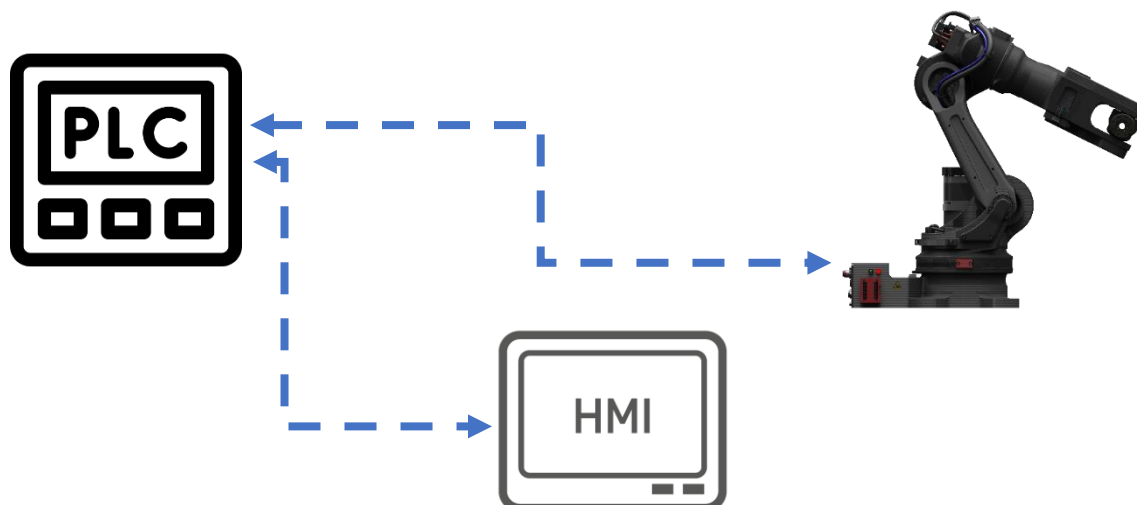
Astorino może działać w dwóch trybach pracy sieci. Robot może być ustawiony jako serwer lub klient. Póła sygnałów pozostaje taka sama.

- Serwer Modbus



## ASTORINO Instrukcja obsługi

- Klient Modbus



### 23.2 Typy obiektów Modbus w Astorino

Następujące typy obiektów mogą być obsługiwane przez serwer Modbus do urządzenia klienckiego Modbus. Adresy są reprezentatywne dla oryginalnej specyfikacji Modicon. Zgodnie z obecnym standardem adres może wynosić 0 - 65535 z typem obiektu identyfikowanym przez polecenie używane do odczytu lub zapisu rejestru. Robot Astorino może odczytywać i zapisywać 3x input register 3x holding register, co daje dodatkowe 56 wejść i 56 wyjść.

Robot storino wykorzystuje standardowy PORT: 502

Jeżeli astorino pracuje jako Serwer

Typ obiektu	Funkcja Astorino	Rozmiar	Przestrzeń adresowa
Input register	Wyjścia	16 bitów	30001 - 30003
Holding register	Wejścia	16 bitów	40001 - 40003

Jeżeli astorino pracuje jako Klient

Typ obiektu	Funkcja Astorino	Rozmiar	Przestrzeń adresowa
Input register	Wejścia	16 bitów	30001 - 30003
Holding register	Wyjścia	16 bitów	40001 - 40003

## 23.3 Konfiguracja portu Ethernet

Ustaw adresy sieciowe zgodnie z konfiguracją PLC/HMI i ustaw ustawienia Ethernet na Modbus TCP Server lub Klient.

Astorino jako Modbus TCP Server

Astorino jako Modbus TCP Klient

Collision det Ethernet Firmware

Ethernet Settings ModbusTCP Server


IP Address  
192 . 168 . 0 . 1

Subnet Address  
255 . 255 . 255 . 0

Gateway Address  
192 . 168 . 0 . 1

DNS Address  
192 . 168 . 0 . 1

Modbus TCP port: 502  Connected

 Save

Collision det Ethernet Firmware

Ethernet Settings ModbusTCP Client

IP Address  
192 . 168 . 0 . 1


Subnet Address  
255 . 255 . 255 . 0

Gateway Address  
192 . 168 . 0 . 1

DNS Address  
192 . 168 . 0 . 1

Modbus Server Address:  
192 . 168 . 0 . 100

Modbus TCP port: 502  Connected

 Save

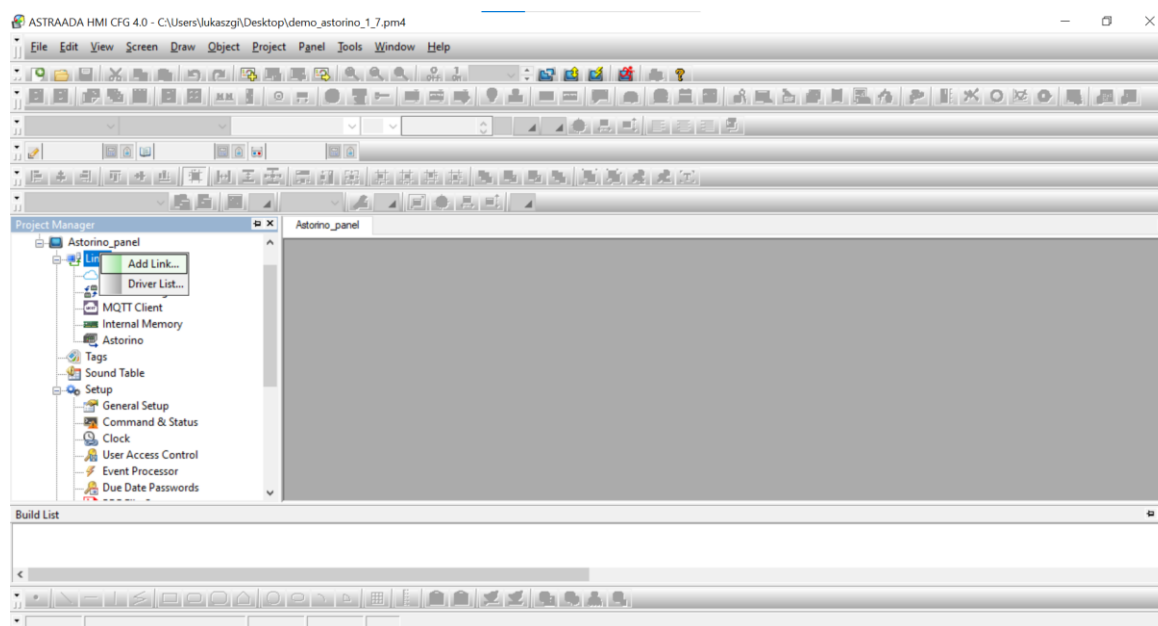
## ASTORINO Instrukcja obsługi

### 23.4 Panel ASTRAADA HMI – przykład

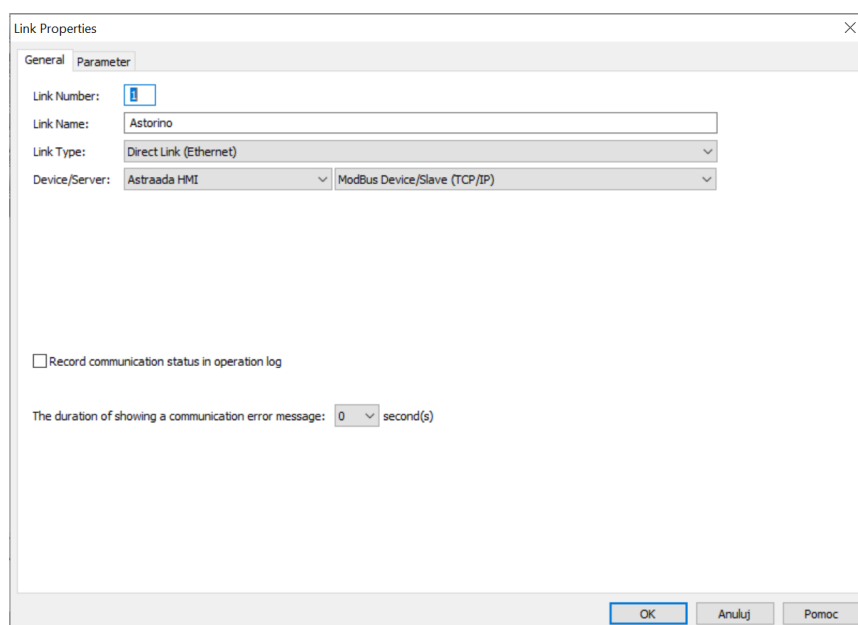
Ten przykład pokazuje tylko konfigurację protokołu komunikacyjnego Modbus TCP na panelach HMI ASTRAADA. Więcej informacji można znaleźć w podręcznikach ASTRAADA HMI. Przykład wykorzystuje robota astorino jako serwer komunikacji.

Otwórz program ASTRAADA HMI CFG i ustaw odpowiedni panel HMI w opcjach.

#### 1. Dodaj link



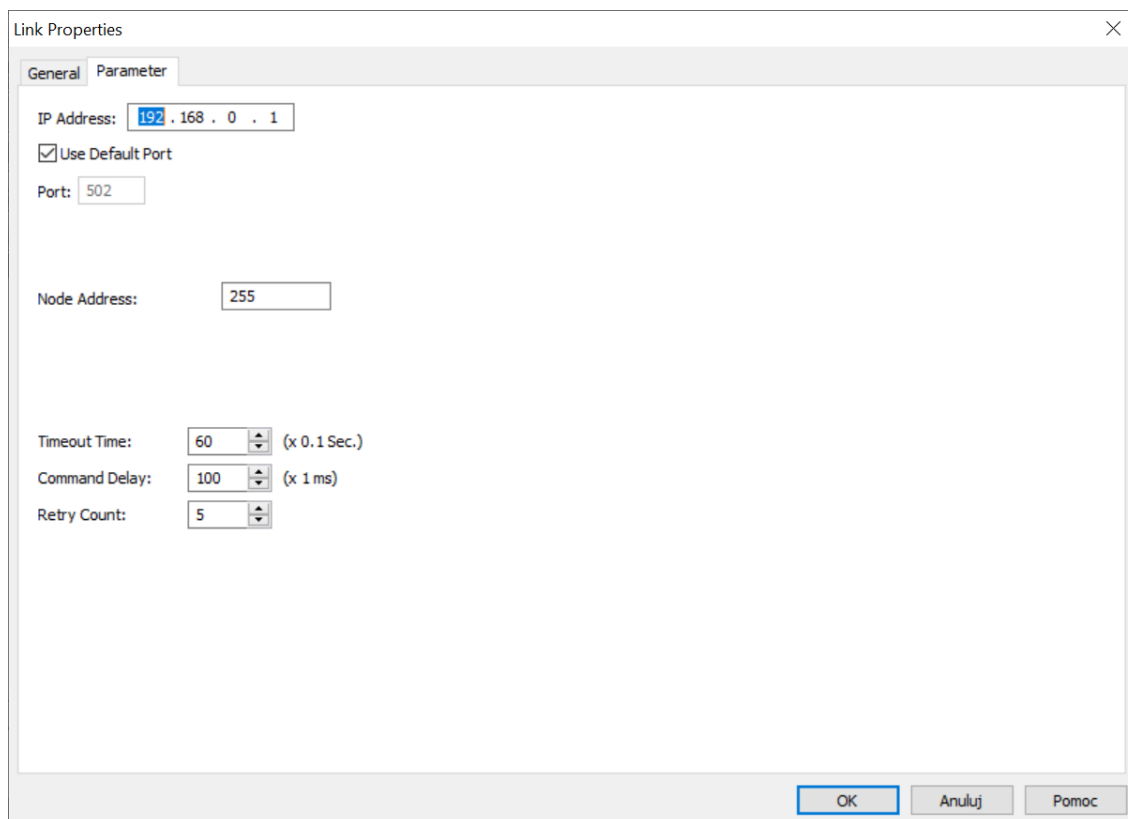
#### 2. Skonfiguruj łącze, dodaj urządzenie ModBus (Device/Slave TCP/IP).



## ASTORINO Instrukcja obsługi

---

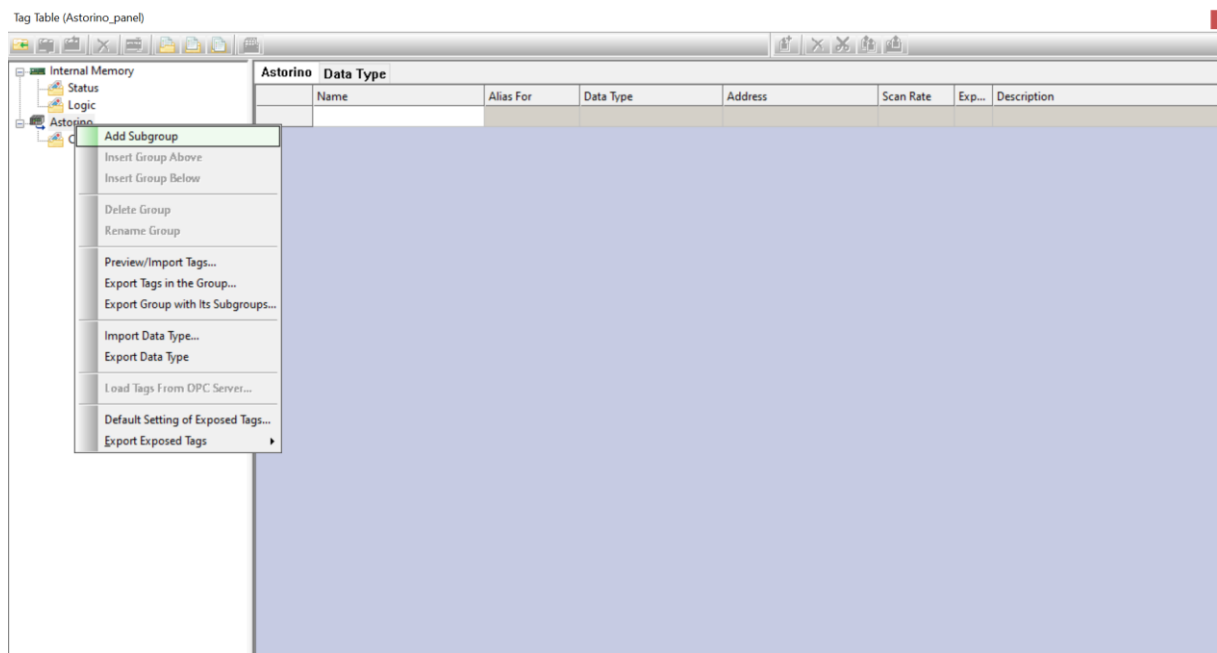
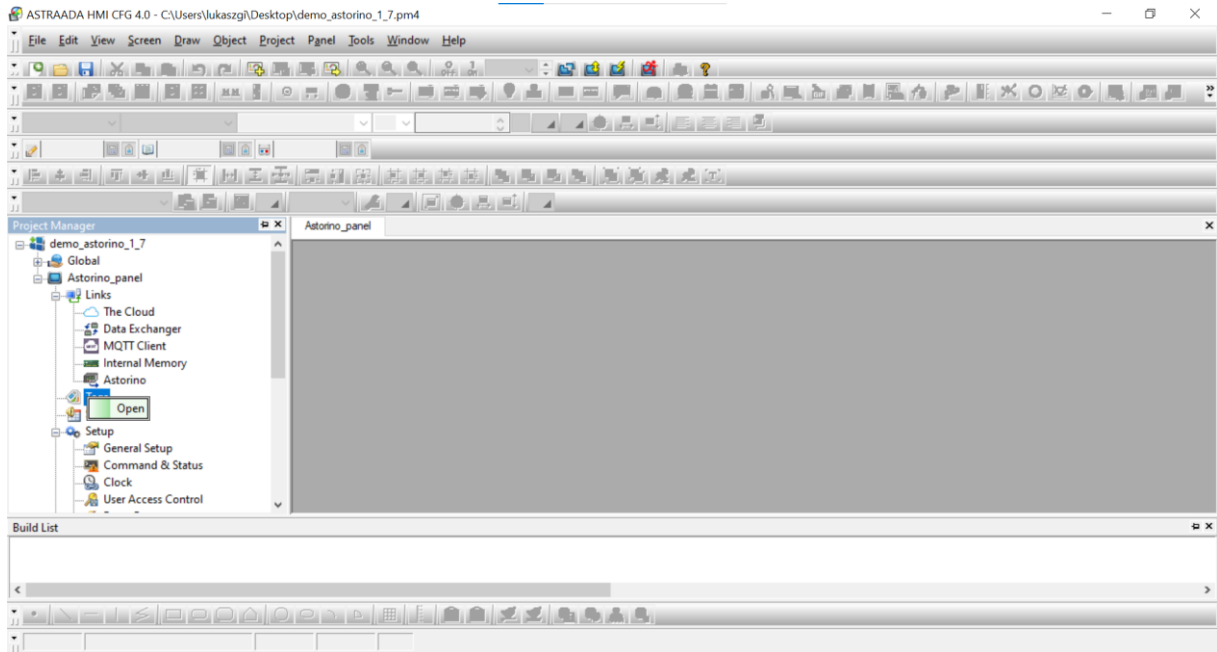
3. Ustaw adres IP, limit czasu, liczbę ponownych prób itp. Zaleca się ustawienie liczby ponownych prób na co najmniej 3, a przekroczenia limitu czasu na co najmniej 3 sekundy.



## ASTORINO Instrukcja obsługi

### 4. Ustaw typ danych

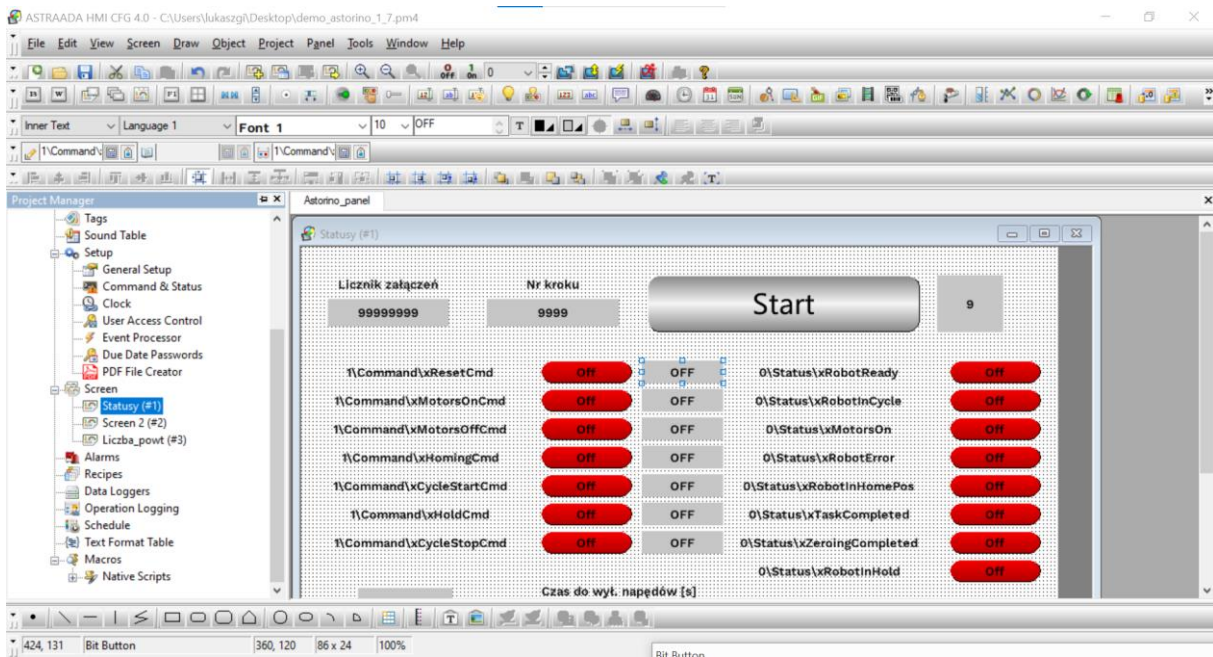
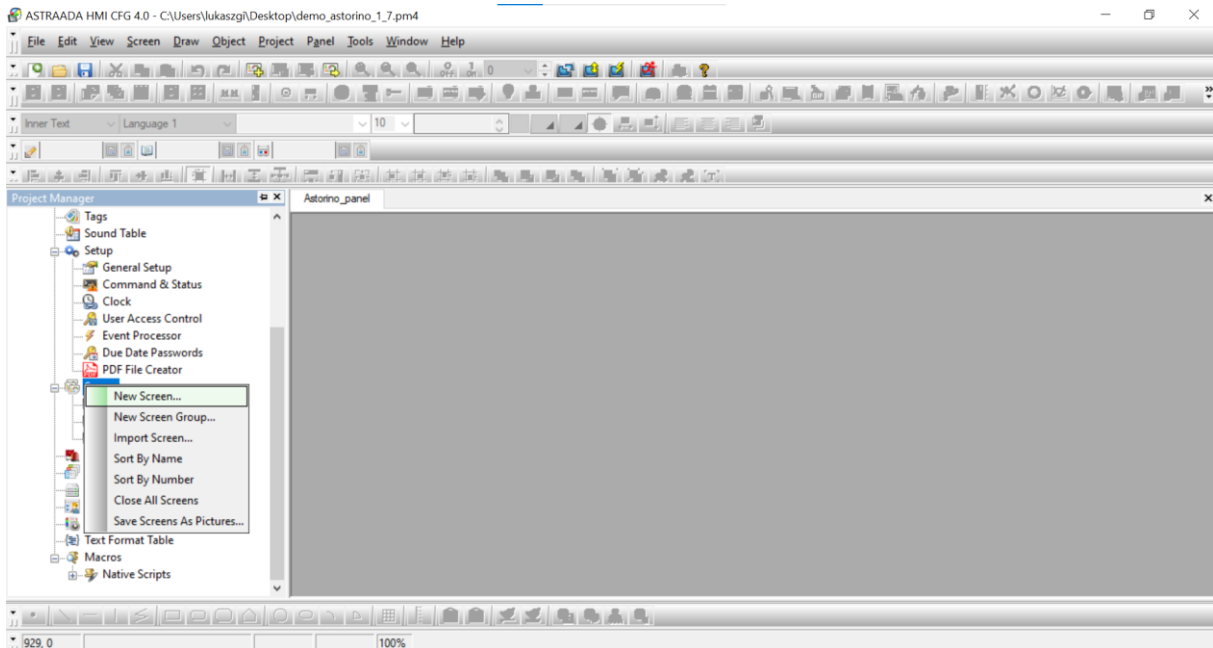
W tym przykładzie wejścia i wyjścia zostaną ustawione jako normalne i de-zykowane.



Command	Data Type						
	Name	Alias For	Data Type	Address	Scan Rate	Exp...	Description
1	xResetCmd		Bit	40001.0	Normal	No	Error reset command on the robot
2	xMotorsOnCmd		Bit	40001.1	Normal	No	Motors on command on the robot
3	xMotorsOffCmd		Bit	40001.2	Normal	No	Motors off command on the robot
4	xHomingCmd		Bit	40001.3	Normal	No	Robot position homing command
5	xCycleStartCmd		Bit	40001.4	Normal	No	Start cycle command on the robot
6	xHoldCmd		Bit	40001.5	Normal	No	Command to put the robot into hold mode
7	xCycleStopCmd		Bit	40001.6	Normal	No	Command to stop the robot's work cycle

## ASTORINO Instrukcja obsługi

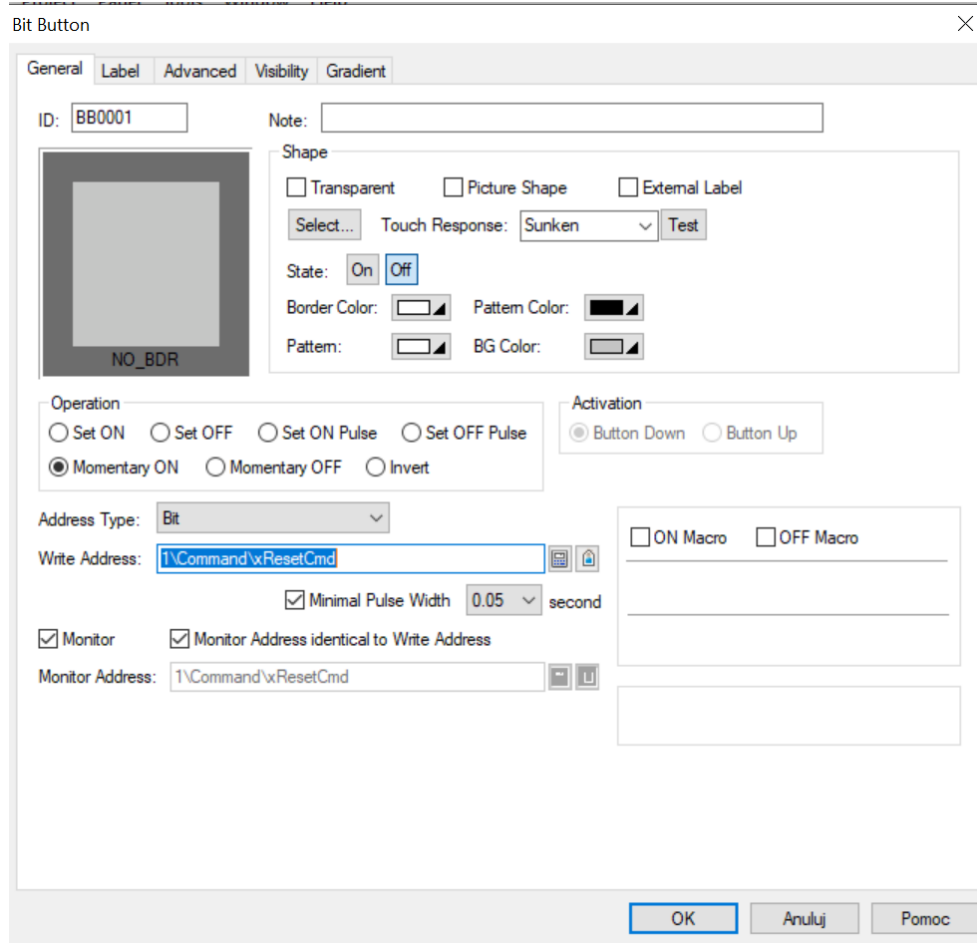
### 5. Dodaj nowy ekran i skonfiguruj jego zawartość.





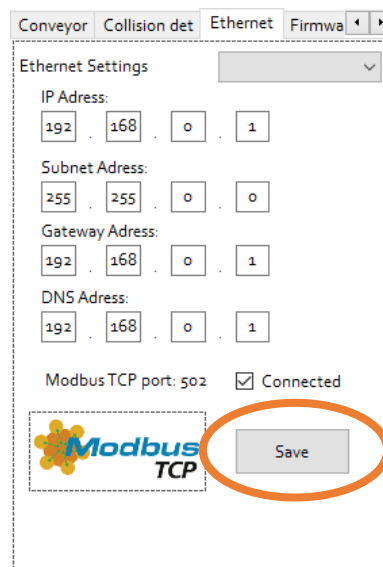
## ASTORINO Instrukcja obsługi

### 6. Skonfiguruj przyciski do odczytu/zapisu poszczególnych adresów.



### 7. Zapisz i wyślij projekt panel HMI.

Jeśli komunikacja działa poprawnie, status zostanie wyświetlony w tym menu.



## ASTORINO Instrukcja obsługi

Stan rejestrów jest wyświetlany na tej karcie. Jeśli bit rejestru jest włączony, to przyciski świecą na żółto.

IO	Modbus	Dedicated IO	Conveyo	◀	▶		
Fieldbus Inputs							
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56
Fieldbus Outputs							
9	10	11	12	13	14	15	16
17	18	19	20	21	22	23	24
25	26	27	28	29	30	31	32
33	34	35	36	37	38	39	40
41	42	43	44	45	46	47	48
49	50	51	52	53	54	55	56

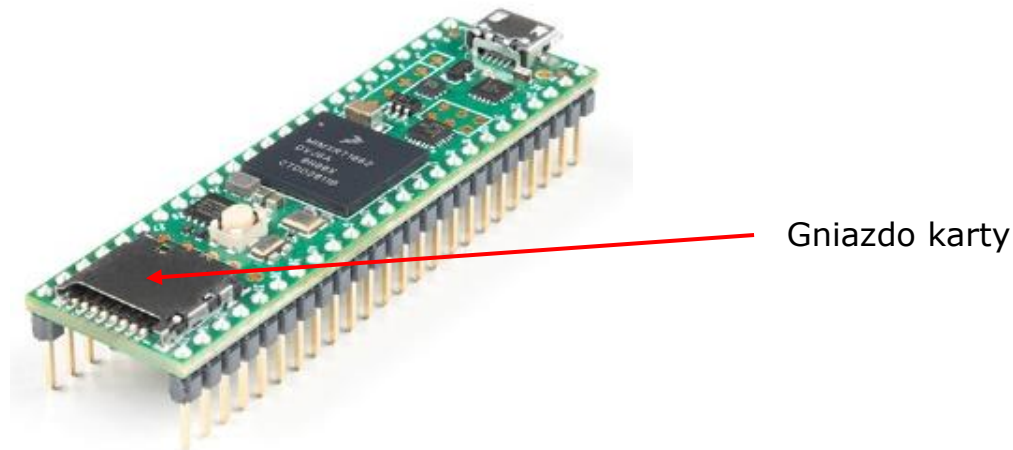
### 23.5 Używanie rejestrów modbus jako wartości rzeczywistych

Do odczytu i zapisu danych liczbowych za pośrednictwem rejestrów Modbus można użyć funkcji **BITS**.

```
.PROGRAM BIT
  BITS 9,16 = 12082 ;sets a number as bits
  x = BITS(1009,16) ;reads a number from bits
.END
```

## 24 Kalibracja

Wykonaj kalibrację po złożeniu robota. Po kalibracji dane zerowania są przechowywane na karcie microSD umieszczonej na głównej płycie procesora wewnątrz podstawy robota:



Oznacza to, że robot nie musi być za każdym razem ponownie kalibrowany, gdy zasilanie jest wyłączone.

Procedura kalibracji jest opisana w podręczniku kalibracji.

## **25 Informacje o producencie**

Kawasaki Robotics Astorino  
INSTRUKCJA OBSŁUGI

---

06.2024: 6. wydanie

Publikacja: ASTOR i Kawasaki Robotics GmbH

---

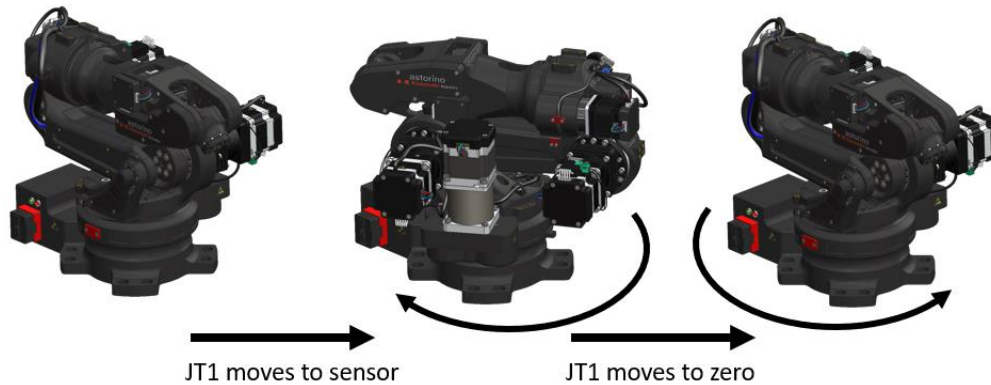
Copyright © 2024 ASTOR & KAWASAKI Robotics GmbH.  
Wszelkie prawa zastrzeżone.

## Załącznik A – Standardowa procedura zerowania

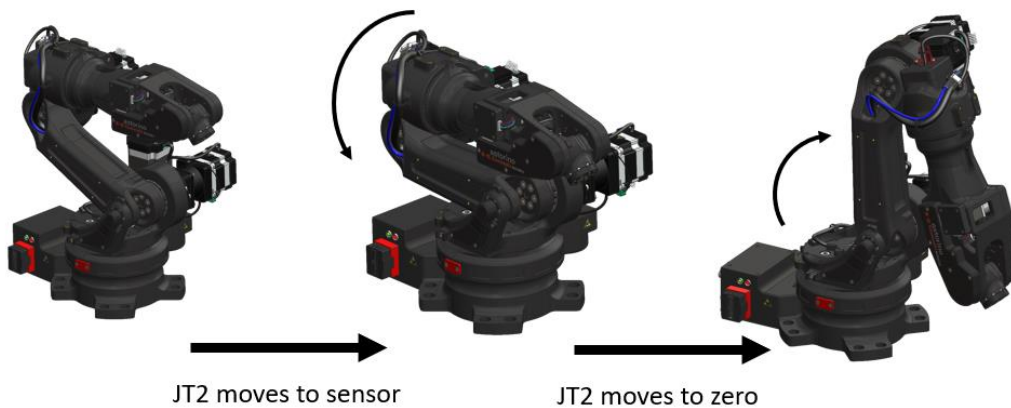
Robot Astorino wyposażony jest w enkodery inkrementalne, dlatego po jego włączeniu wszystkie osie muszą zostać wyzerowane.

Ta procedura jest automatyczna i w domyślnej konfiguracji jest opisana poniżej.

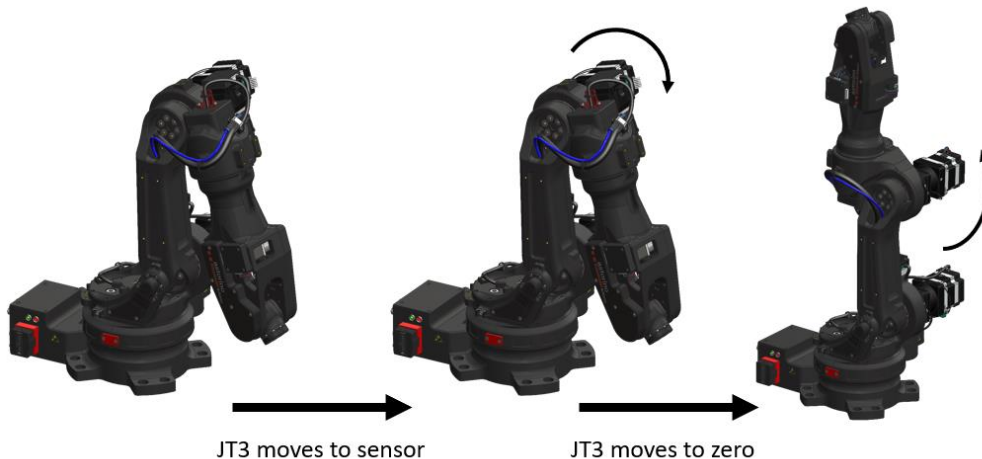
JT1:



JT2:



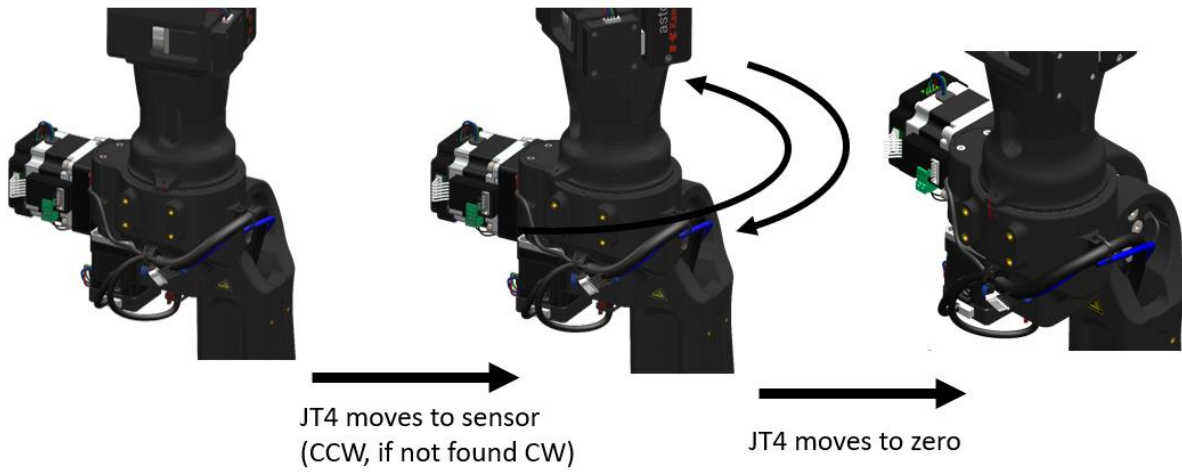
JT3:



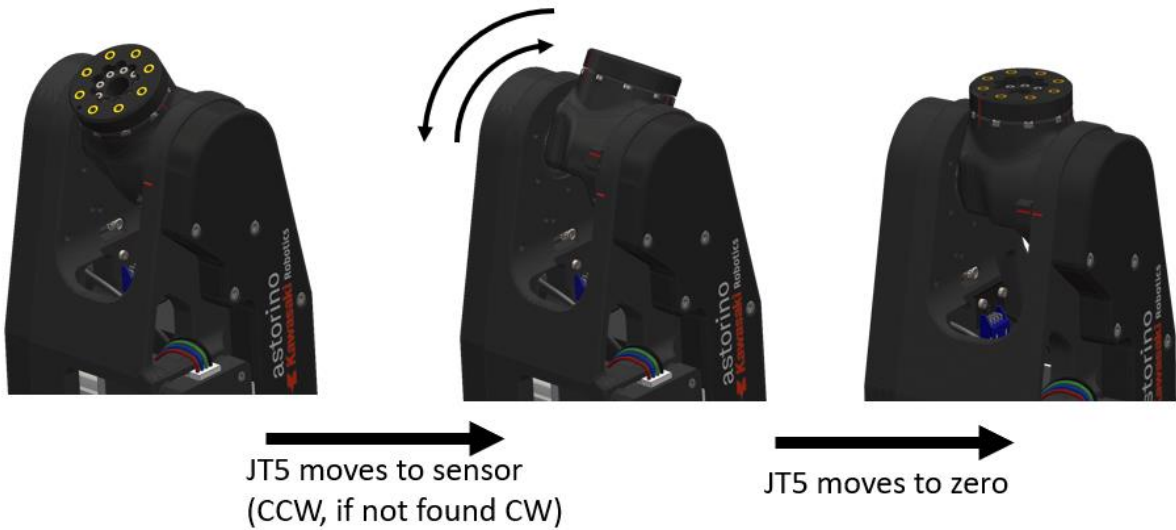
## ASTORINO Instrukcja obsługi

---

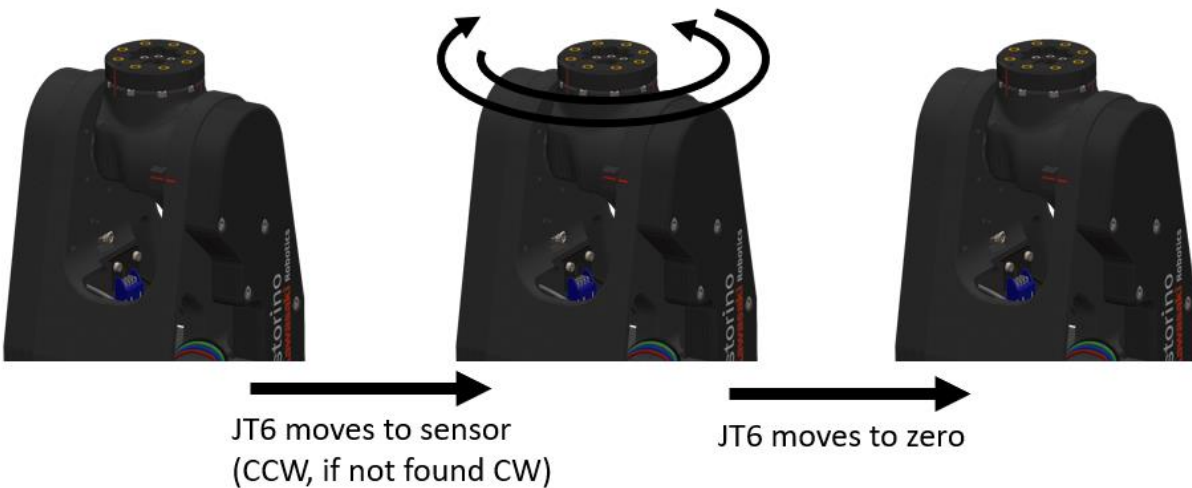
JT4:



JT5:



JT6:



## Załącznik B – materiał PET-G

PET-G jest jednym z najbardziej wszechstronnych materiałów do drukowania 3D, mocnym i łatwym do wydrukowania. Jego popularność wzrosła w ciągu ostatnich lat jako alternatywa dla PLA. PET-G to wariant PET stosowany w druku 3D. G na końcu oznacza zmodyfikowany glikolem. Zmiana ta wpływa na strukturę chemiczną, dzięki czemu materiał jest bardziej przezroczysty, mniej delikatny i łatwiejszy w obróbce.

PET-G ma bardzo ciekawe właściwości, a jego najbliższymi konkurentami są PLA i ABS. Główne właściwości, które należy wziąć pod uwagę, są następujące:

1. **Sztywność:** Trudność w deformowaniu materiału, w tym rozciąganie i zginanie. PLA jest bardziej sztywny niż PET-G, PET-G i ABS są prawie tak samo sztywne.
2. **Odporność:** PET-G jest na ogół trudniejszy do złamania niż PLA i ABS. W oparciu o właściwości techniczne, PETG jest nie tylko bardziej odporny niż ABS, ale przyczepność między warstwami jest wyższa, co daje ogólnie lepszą odporność.
3. **Odporność na ciepło:** PET-G mięknie w temperaturze 80°C, podczas gdy PLA może zacząć mięknąć w temperaturze 50°C. Jednak ABS ma najwyższą odporność na ciepło, mięknięcie w temperaturze 105°C.
4. **Drukowanie bezwonne:** W przeciwieństwie do ABS, PET-G nie wytwarza zapachu podczas drukowania.
5. **Nadaje się do recyklingu:** Ze względu na swoją popularność, większość miast ma wymaganą infrastrukturę do recyklingu PET-G.

### Jak wydrukować PET-G

Temperatura hotendu: PET-G jest zwykle drukowany w temperaturze 220-250°C i może być drukowany za pomocą prawie każdej drukarki 3D, w tym całkowicie metalowych hotendów lub tych, które używają wewnętrznej rurki PTFE.

1. **Temperatura powierzchni:** Aby wydrukować PET-G, konieczne jest użycie podgrzewanego stołu w temperaturze 60-90°C. Zaleca się również dodanie kleju, takiego jak klej do papieru, do powierzchni wydruku.
2. **Zamknięte drukarki 3D:** Mimo że nie jest konieczne używanie zamkniętych drukarek 3D, zalecamy unikanie wahań temperatury pokojowej.
3. **Wentylator warstwowy:** Zaleca się używanie wentylatora warstwowego podczas drukowania PET-G.
4. **Wypaczanie:** PET-G ma zmniejszony skurcz termiczny, więc nie jest podatny na wypaczenia i powoduje powstawanie części o dobrych tolerancjach wymiarowych

## Załącznik 3 – podłączenie typu PNP

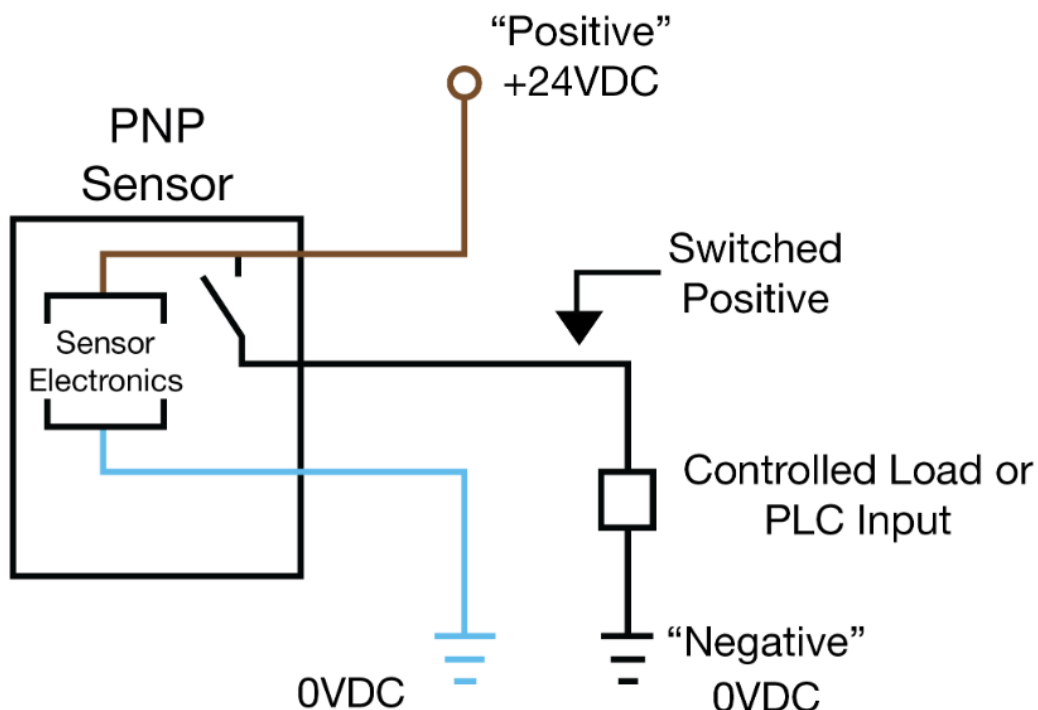
PNP oznacza Pozytywny, Negatywny, Pozytywny. Znany również jako sourcing. Na module IO wejście PNP, gdy nie jest sterowane, jest podciągnięte do stanu wysokiego, np. +24V.

Najczęściej w Europie jest typ wejścia/wyjścia PNP, który będzie używany z czujnikiem PNP lub siłownikiem. Mniej powszechne w dzisiejszych czasach są karty wejściowe NPN, które były popularne w Azji i wymagają czujnika typu NPN do poprawnego działania.

Oto prosty sposób na zapamiętanie, jak podłączyć 3-przewodowe urządzenie PNP:

PNP = Przełączany dodatni

"Przełączany" odnosi się do tego, która strona kontrolowanego obciążenia (przełącznik, mały wskaźnik, wejście PLC) jest przełączana elektrycznie. Albo obciążenie jest podłączone do ujemnego, a dodatnie jest przełączane (PNP).



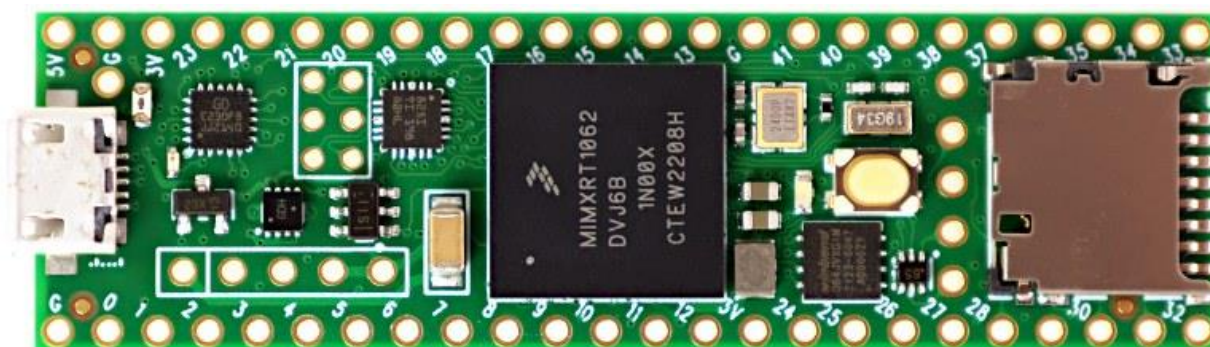


## Załącznik 4 – Teensy 4.1

Teensy 4.1 to najpotężniejszy mikrokontroler kompatybilny z Arduino dostępny obecnie na rynku. Oparty na NXP i.MX RT1062 ARM Cortex-M7 pracujący z częstotliwością 600MHz z możliwością przetaktowania. Być może najlepsze jest to, że jest kompatybilny z popularnym środowiskiem programistycznym Arduino IDE, a także z wieloma istniejącymi bibliotekami Arduino, dzięki czemu jest bardzo łatwy do uruchomienia, w przeciwieństwie do wielu innych zaawansowanych mikrokontrolerów, które są dostępne.

Sercem mikrokontrolera i.MX RT1060 jest rdzeń procesora ARM Cortex-M7, który wnosi wiele zaawansowanych funkcji do prawdziwej platformy mikrokontrolerów czasu rzeczywistego.

Cortex-M7 to podwójny superskalarny procesor, co oznacza, że M7 może wykonać dwie instrukcje na cykl zegara, przy 600MHz! Oczywiście wykonanie dwóch jednocześnie zależy od instrukcji i rejestrów zamawiających kompilator. Wstępne testy porównawcze wykazały, że kod C++ skompilowany przez Arduino IDE ma tendencję do osiągania dwóch instrukcji na cykl około 40% do 50% czasu podczas wykonywania numerycznie intensywnej pracy przy użyciu liczb całkowitych i wskaźników.



Więcej informacji można znaleźć na stronie internetowej PJRC.

<https://www.pjrc.com/store/teensy41.html>